Universidad Carlos III de Madrid

BACHELOR'S THESIS

BACHELOR'S DEGREE IN INDUSTRIAL TECHNOLOGIES ENGINEERING

Modelling and control of a UAV-USV collaboration scheme for fluvial operations

Author
Omar Velasco Anrar
Director
João Ricardo Pereira Valente

Polytechnic School
Systems Engineering and Automation

February 27, 2017

To my mother for the education she gave me
To my father for the knowledge he gives me
To them and my friend/s for always being there

Abstract

This thesis focuses on a Model Based Design approach to the dynamic modelling and control design of a multi-robot solution based on a collaboration scheme between a UAV and USV. The purpose of the system is to provide a suitable platform to autonomously perform limnology related surveys.

The dynamic models of both platforms are derived from a Newton-Euler formalism and implemented through block oriented modelling using the Simscape Multibody toolset within Simulink. The implementation of both the simulation architecture and the control architecture are described and explained. This control architecture is based on PID feedback loops that are used for achieving control of the UAV and USV dynamics. Finally, the built simulator is used to asses the performance and reliability of the designed controllers and the dynamic modelling approaches selected.

Contents

List of Figures

List of Tables

1 Introduction

This chapter gives an overview of the topics that will be covered in this thesis. It starts with a description of the motivation of the project and follows up with an analysis of the contribution to be made by this work. The objectives to be accomplished by this work are also covered. A short layout of the chapter structure of the thesis is also given.

1.1 Motivation

Limnology, is the study of inland waters, lakes, reservoirs, rivers, wetlands, groundwater, etc. as ecological systems and their interaction with their drainage basins and the atmosphere. It is considered a part of ecology and it covers the biological, geological, physical, chemical and other components of inland aquatic ecosystems. It integrates this components along with the analysis of the drainage basin and movements of water through it, the interaction with the atmosphere and biochemical changes that occur en route [11].

Recently a sub-discipline of limnology, called landscape limnology, uses a landscape perspective to study, manage and conserve aquatic inland ecosystems. According to the definition given by [12], landscape limnology is the spatially-explicit study of lakes, streams, and wetlands as they interact with freshwater, terrestrial, and human landscapes to determine the effects of pattern on ecosystem processes across temporal and spatial scales.

Limnology studies are based on the recollection of data from the study site and traditional field survey methods quickly lose their appeal as much faster and less laborious data collection techniques based on autonomous procedures are developed. Unmanned aerial and surface vehicles (UAV and USV) are making their way into the field by providing access to new technology approaches to data recollection along with the augmented accessibility provided by these platforms.

Remote sensing techniques are becoming increasingly popular as they can be used to access dangerous sites or places where traditional manned sensing techniques are impossible to implement. Besides this, autonomous platforms tend to provide a permanent record of the measured conditions which makes revisiting and comparison over time possible.

Unmanned aerial vehicles, are one of the most used platforms in this new development of the field, being multirotor vehicles (generally quadcopters or hexacopters) the more outstanding solutions. The presence of multicopters in research fields is not something unusual as the specifications of the on-board equipment keeps improving and new application opportunities arise. This development, that can also be seen in commercial and military fields along with research based studies, hints to the increase in the suitable applicability of UAV systems around us.

Multicopter platforms are of particular interest due to several advantages they provide. The capability of multicopters to perform vertical take off and landing (VTOL) is one of their most relevant features, since there is no re-

quirement for large landing and take off platforms. Along with this, the use of several rotors implies a smaller rotor diameter than an equivalent helicopter rotor, opening the possibilities to rotor enclosure and reducing the amount of kinetic energy stored in the individual propellers during flight. This allows safe operation (for both the vehicle and the surroundings) in indoor or obstacle dense environments. From the control point of view, multicopter systems rely on the individual control of their rotors in order to achieve the desired actuation. This presents a big advantage over complex rotor actuators like the ones found in helicopters, simplifying in a notable manner both the design, control and maintenance of the platform. Besides this, their hovering capabilities and high stability control make multicopter vehicles a good platform for precise operations.

One of the main drawbacks of current multicopters is their flight autonomy. Although this problem is being mitigated with the incoming rush of higher capacity batteries, it is still one of the main limitations of the platform. Flight autonomy limits the reach of the vehicles and hinders the capability of deployment on large or difficult-access areas.

Here is where multi-robot platforms come at play. Multi-robot solutions rely on the use of more than a single robot operating on the same environment to perform more efficiently the required tasks. This translates into more robust and reliable systems in tasks benefited by modularization and improves the effectiveness in terms of performance thanks to the collaborative effort of the platforms.

Many of the multi-robot solutions present in recent literature such as [13], [14] or [15] rely on heterogeneous systems to perform different tasks. In this thesis, a collaborative scheme between an UAV and a USV is suggested. The addition of a USV to the autonomous system provides the robustness and autonomy the aerial platform lacks, making up for the deficiencies found in typical quadcopter solutions.

While many of the more advanced USV applications are involved in the military field, they have started to make their way into the commercial and research grounds. USV are really valuable in marine and limnology studies as they provide more flexibility than fixed survey buoys and do not incur in the high cost of weather ships or research vessels. USVs are able to have months of on-site persistence as they usually are capable of harnessing their own energy being this one of the most notable advantages among others.

1.2 Contribution

This work presents a collaborative scheme between an autonomous surface vehicle and an autonomous aerial vehicle. The selected aerial platform is the AR.Drone quadcopter while the surface vehicle, the Strider V1.0, is a prototype platform developed specifically for this project [3].

The objective of this multi-robot solution is to provide a collaboration sys-

tem to perform data collection tasks from calm water environments in the frame of limnology studies. The Strider V.1 provides with the longer autonomy lacked by the aerial vehicle while the drone provides the agility and flexibility to the system, making up for this deficiency, inherent to most surface vehicles.

The USV turns up to be a mobile landing platform for the drone where on deck charging of the UAV battery can be done. The UAV's flexibility of operations allows the access to zones where the surface vehicle cannot reach, along with different types of aerial surveys, now trending on the remote sensing community. While the USV performs this "piggybacking" duty, it can also be equipped with several surface based survey equipment to complement and expand the data pool. The autonomous operation of the complete system could provide a continuous stream of precise data to be used in further analysis or live monitoring.

1.3 Objectives

The main objective of this work is to obtain the dynamic models and develop a suitable control for both the UAV and the USV. In order to do this MATLAB/Simulink and the latest Simscape Multibody workshop release will be used, this allows for accurate mechanic implementation of imported CAD models and to obtain realistic dynamic simulations while simplifying the formulation and computation of the complete mechanical systems.

Once the dynamic modelling of both vehicles is complete, a control system based on PID feedback loops is implemented to perform the required tasks for the systems. Way-point guidance control will be used for path following on the Strider V1. The aerial vehicle will have a position control to achieve autonomous vertical take of and landing, hovering over the mobile platform and path following and escort duties for the surface platform.

This thesis will provide with the necessary control laws that will need to be implemented in the working platforms. To do so, this work will take up onto the beginning phases of a Model-Based Design (MBD) process. MBD is a mathematical and visual analysis approach for developing complex design control, signal processing and communication systems [16, 17]. MBD is usually divided in four steps:

1. Plant modelling
2. Controller design
3. Offline or Real time Simulation
4. System integration

This list of processes will be worked up to the offline simulation step. The advantages brought by MBD are the possibility of quick prototyping, rapid design and parameter changes. Also high efficient and fast deployment through solutions like hardware in the loop testing make up good arguments for this design method.

1.4 Thesis layout

This thesis is structured as follows:

Chapter 2 overviews the state of the art of autonomous vehicles, multi-robot systems, simulation of both UAV and USVs and works on the simulation environment to be used during this thesis.

Chapter 3 deals with the regulatory frame concerning autonomous vehicles, in particular drone flight in the frame of the Spanish legislation and USV legislative treatment. It also provides with a short brief into the socio-economic impact and the motivation of this work.

Chapter 4 delves into the dynamics of the two platforms to be studied. This chapter provides with the necessary theoretical background and the basic notions of the dynamic model of both quadrotors and vessels. It also provides with the mathematical expressions for the dynamics of the systems.

Chapter 5 presents the simulation environment to be used, the system architecture and implementation in the simulation environment and the parametrization of the simulation.

Chapter 6 presents the control approach taken in this work. It gives an overview of PID control techniques and illustrates the control architecture used for achieving the autonomous operation of the system.

Chapter 7 shows the simulation results of different tests carried out in order to asses the performance of the designed control architecture.

Chapter 8 evaluates the results of this thesis and considers the ongoing and future work to improve the multi-robot platform, the simulation environment and the control system.

Appendix A covers the topics concerning thrust and torque generation of propellers due to aerodynamic forces along with other aerodynamic effects relevant to quadcopter fight.

Appendix B expands some of the concepts on hydrodynamic modelling covered in chapter 4. It also deals with the modelling of hydrodynamic forces and the concept of hydrodynamic derivatives, as well as their calculation.

2 State of the art

In the last few years the development of autonomous vehicles has exploded into a mainstream trend, being UAVs one of the most noteworthy fields along with autonomous cars. It is their capabilities, their flexibility and their performance what makes them so valuable. This trend has made UAVs more accessible to research, commercial and military uses. Thanks to this, many applications such as naval operations have been pushing its limits in increasingly difficult settings over the last years. Some works such as [18] deal with the problems involved in the utilization of UAVs in maritime applications, while others like [19] and [20] provide excellent examples of the applicability of USVs in oceanic environments.

However, the most relevant works for this thesis are the ones that cover the application of UAVs in fluvial surveying, which can be applied to the fields of hydraulics, river morphology, mapping, holistic data collection, etc. This can be seen in works such as [21] where UAV photography-based methods are used to create a digital bathymetry model of river channels. Casado et al. [22] presents with an autonomous method for the identification of relevant hydromorphological features in the frame of quality assessment of riverine ecosystems based on aerial imaging. Some recent works of Amy Woodget and others [23] and [24] delve into the technological advances of utilization of drones as a reliable and accurate method for remote sensing.

On the same note, USV platforms have been used independently for different applications in maritime and fluvial missions, although much less frequently on the latter. Unmanned vehicle utilization in search and rescue mission is a recursive topic as can be seen in [25] where successful preliminary testing of human target tracking in marine search and rescue mission contexts is presented. Another more recent example of this is given by Huang et al. [26] where the design of an autonomous trimaran for maritime rescue is tackled, from the integrated control architecture to the communication network implementation. Siyang and Kerdcharoen [27] present with an approach to inland water inspection made efficient by the use of mounted sensory equipment in a USV. Collaborative USV applications can also be found in [28] where two autonomous surface vehicles work together in automated oil skimming operations for spill clean up and contention.

This huge ongoing development in the autonomous vehicles field has opened the door to multi-robot systems and collaborations schemes between different platforms. Many of the multi-robot solutions present in recent literature like [13], [14] and [15] rely on heterogeneous systems to perform different tasks.

Concerning the type of multi-robot systems pertaining to this work, the community has quickly caught up on the collaboration of both UAV and USV due to advantages this association brings. This can be seen in [29], that presents one of the first UAV-USV robot teams used for emergency situations after the landfall of hurricane Wilma (2005). Also in the frame of large scale disasters, Matos et al. [30] presents a maritime search and rescue UAV-USV system in the frame of the euRathlon 2015 competition. The work of Djapic et al. [31] addresses the design of a collaboration system where the USV is used for the transportation

of both UAVs and UUVs (Unmanned Underwater Vehicles) to remote marine locations and a coordination and cooperation network is developed between the different robots to achieve autonomous VTOL of the UAV from the USV.

Although it is relatively easy to find this kind of multi-robot collaboration schemes between UAVs and USVs, the large majority of this type of collaborations present solutions for maritime environments and applications. It is difficult to find this combination of multi-robot system application on fluvial or inland water environments.

Only three similar projects to this work can be found in the literature. Apart from the aforementioned work of Djapic et al. [31], Mancini et al. [32] shows in his work the application of a UAV and USV for high resolution mapping of inland water areas, however the results obtained from the different platforms were analysed and merged after the scanning and no coordination efforts between the platforms were made during the scanning activity.

The work of Pinto et al. [33] involves a more complex collaboration system designed for environmental data gathering. In this multi-robot system, called RIVERWATCH, the UAV actuates expanding the system's awareness of the environment providing the USV with an augmented perception. This allows the complete system to improve navigation cost, safety and efficiency of data collection.

As it can be seen from this previous examples, UAV-USV collaboration schemes present a wide arrange of different levels of complexity. As Farinelli et al. [1] proposes in his work, multi-robot systems can be classified according to the taxonomy chart seen in Figure 1. As the categorization deepens in the chart, it becomes harder to find UAV-USV collaborative approaches to problem solutions. This is easy to see analysing the mentioned literature above. While cooperative unaware systems are among the most common found collaborations, aware and coordinated are more infrequent, especially in the particular robot team setup pertaining this work.

Concerning the design of the simulation, the works on the application of PID control techniques to quadcopter command are widely available. Some examples of this are [8], [34] or [35], where PID techniques are successfully applied to both simulated and real system quadrotor controllers. Some works such as [36] and [37] are even specific about the platform used in this work, the AR.Drone 2.0. Both of these works deal with the modelling and experimental system identification of the quadcopter and present simulations made in the MATLAB/Simulink environment using the transfer functions obtained during the study.

PID control techniques have been also successfully applied to course and speed control in autonomous surface vehicles as shown in [38], [39] and [33]. Beinset and Blomhoff [38] in particular show a transfer function based Simulink simulation for control design and later applies the results obtained from this simulation to the real platform with favourable results

The appearance of Simulink in this research field is a common occurrence as

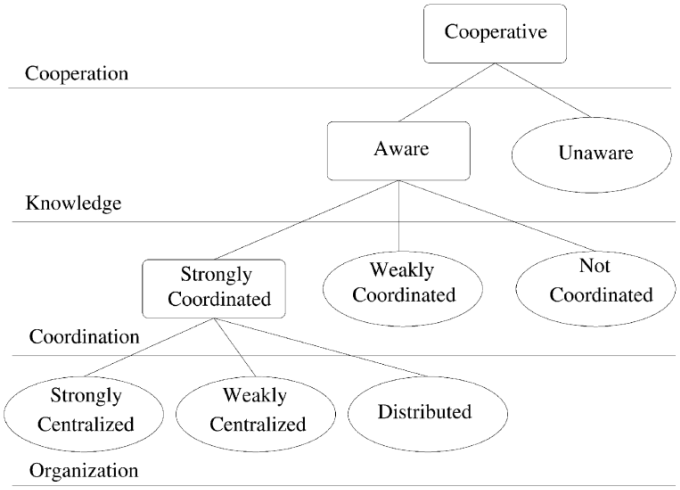


Figure 1: Multi-robot system taxonomy. From [1].

the aforementioned literature shows. Simulink is a renowned tool in the research community and it has been used many times in simulation related works. However, when dealing with the newly available toolsets, due to the quick evolution of Simulink workshops, it is difficult to find works within the latests Simscape toolset environment. Some works within this environment present simulations in several fields dealing with electronics [40], mechatronics [41], robotics [42] and even cardiovascular control [43].

3 Regulatory Framework and Socio-economic Impact

This chapter will give a short brief of the regulatory framework for autonomous vehicles and the socio-economic impact of the application of the multi-robot system presented in this work, this impact is highly related to the motivation of the project. It will end up giving a budget estimate of the project.

Regulatory framework

Legislation on autonomous vehicles is hardly catching up to the rapid development of autonomous vehicles. As new developments in the autonomous vehicle field are being made, regulatory frameworks are slowly being put in place to legislate this new area. Autonomous vehicles present with an unprecedented legislative and philosophical debate that keeps evolving along with this new technology due to the challenges it presents.

In Spain, a specific regulation was put up in 2014 to establish a series of urgent measures to control the operation of unmanned aerial vehicles under 150 Kg. Although a new draft for a legislation update was presented at the UNVEX 2016 conference, expanding the scope of the previous law, the current legislation on UAV flight is still the Real decreto 552/2014 [44] issued in 2014. The law categorizes the drones in three mass ranges, under 2 Kg, between 2 Kg and 25 Kg and over 25 Kg. Each of the categories has different range and altitude restrictions. Regardless of the category, all drones must be marked with an ID plaque and an official license issued by the Spanish State Aviation Safety Agency (AESA) is mandatory for UAV flight. Restrictions on controlled airspace areas are also applicable to drone flight, requiring additional permission from AESA with at least five days prior to flight. Similar conditions and restrictions can be found in other regulatory frameworks issued by aviation agencies such as the EASA (European Aviation Safety Agency) or the American FAA (Federal Aviation Administration).

While aerial legislation quickly dealt with unmanned vehicles due to the inherent risks and importance of airspace control, surface and underwater autonomous vehicles do not seem to fit any particular description in the current Spanish legislation. Due to this it is extremely difficult to find a unitary legal framework due to the wide array of characteristics this type of vehicles feature. Although this issue has been noted, it is still being worked on. This same legal void can be found in many other country legislations.

Socio-economic impact

Great development has been done in the past few years on remote sensing techniques for several limnology related surveys, in special data collection techniques based on aerial platforms for limnology studies such as bathymetric LiDAR or high resolution aerial imagery. Progress on USV based techniques has

also shown an improvement on speed and quality of data recollection methodologies.

This development is driven off by the implantation of new European environmental legislations with the aim of improving the quality of inland water ecosystems. As ecosystem protection issues flare concern over water quality and ecology, accurate and rapid assessment of inland water environments becomes necessary to success in the task of providing up to date data for addressing environmental hazardous situations.

The multi-robot system presented in this work seeks to be the starting point to a future standalone platform for deployment of this techniques providing the necessary system structure and architecture to perform this surveys autonomously. This is a really valuable improvement in habitat assessment as the availability of freshwater dwindles and control of this vital resource becomes a paramount issue in ecosystem preservation. Besides, autonomous operation unlocks the possibility of obtaining live data from particular areas of interest without human deployment to the study site.

Budget estimate

The budget estimate in Table 2 shows an estimate of the cost of the development of this project. This cost is related to the development of the MBD process workflow, namely plant modelling, controller design and offline simulation. The budget is divided into the cost of the used software and the labouring hours put into the project from start to finish.

The total duration of the project was six months, 25 weekly hours of work will be assumed to consider the labouring costs.

Description	**Quantity**	**Unit price**	**Amount**
Working hours	600 h	10 €	6000 €
Total			6000 €

Table 1: Labour cost.

Concerning the required software licenses, a Matlab Educational class license is selected for the estimation of the cost of the software due to the research character of this work. The following Matlab packages have been used in this work:

- Matlab 2016a
- Simulink
  - Simulink Control Design
  - Simscape
    - Simscape Multibody

– Simulink 3D Animator

Description	**Quantity**	**Unit price**	**Amount**
Educational MATLAB license	1	500 €	500 €
Educational Simulink license	1	500 €	500 €
- Simulink Control Design license	1	200 €	200 €
- Simulink 3D Animator license	1	200 €	200 €
- Simscape license	1	200 €	200 €
- Simscape Multibody license	1	200 €	200 €
Total			1800 €

Table 2: Software cost.

After considering VAT rates the following total cost for the project is obtained:

Description	**Amount**
Software cost	1800 €
Labour cost	6000 €
Net total	7800 €
VAT rate (21%)	1638 €
Total	9438 €

Table 3: Total budget estimate.

4 Dynamic modeling

This chapter deals with the derivation of both the UAV and USV dynamic model. This describes how the systems react according to its different inputs. Through this equations it will be possible to correctly define the expected simulation model to obtain accurate results.

The first section (4.1) will deal with the dynamics of the quadcopter while the second one (4.2) deals with the Unmanned Surface Vehicle. Both sections give an introduction to the platform and cover basic concepts and their Newton-Euler model. In this work, the Newton-Euler formalism and the Euler angles theory have been chosen.

4.1 Unmanned Aerial Vehicle: AR.Drone 2.0

This section will deal with the Unmanned Aerial Vehicle to be simulated, the AR.Drone 2.0. The first part of this section will present the multicopter that will be used in the project. The second one delves into the basic concepts of quadrotor flight while the third one deals with the Newton-Euler model of the quadrotor.

4.1.1 The AR. Drone 2.0

The multicopter to be used in this work is the AR. Drone 2.0 from the French company Parrot. The AR. Drone is an electrically powered quadcopter. This quadcopter, designed for entertainment, has been widely used in many scientific researches thanks to its characteristics at his low price point

One of the key features of this platform is its open and well documented API, along with a huge online community, allowing easy access to the data from the built-in sensors and the images from its cameras, making it a really useful platform for researchers. This makes for quick proof-of-concept experiments and ease of implementation in the platform.

Figure 2: AR.Drone 2.0. From [2].

The quadcopter features a durable carbon fiber support structure, plastic body, sensor and control board, two indoor and outdoor removable hulls, two cameras and four high efficiency brushless motors. It can achieve speeds over 5 m/s and has an estimated flight time of 13 minutes with one charge, although higher capacity batteries are available in the aftermarket.

The sensory equipment of the drone consists of a 6 degrees of freedom IMU (inertial measurement unit), a sonar based altimeter, a barometer and the two aforementioned cameras (vertical and frontal) besides a GPS unit (only included in the GPS edition). Some of the characteristics of the quadcopter are listed in Table 4.

	Element	Value/Precision
Sensors	3 axis Gyroscope	2000 $/s$
	3 axis Accelerometer	$\pm$ 50 mg
	3 axis Magnetometer	$\pm$ 6
	Barometer	$\pm$ 10 Pa
	Ultrasound sensor	$Maximum\ altitude$: 6 m
	GPS	$\pm$ 2 m
Frontal Camera	Image Resolution	1280x720 px
	Video Resolution	720p
	Frame rate	30 fps
Vertical Camera	Image Resolution	320x240 px
	Video Resolution	$QVGA$
	Frame rate	60 fps
Processing	Processor	1Ghz 32 bit ARM Cortex A8
	RAM	1GB DDR2 200 mHz
	SO	Linux 2.6.32
Battery	Capacity	1200mAh
	Autonomy	$8 - 12$ min
	Technology	Lithium polymer
Total mass	With indoor hull	420 g
	With outdoor hull	380 g

Table 4: AR.Drone characteristics.

4.1.2 Quadcopter modelling

A quadcopter is a multirotor copter lifted by four equally spaced rotors in two pairs spinning in opposite directions, generally arranged symmetrically in a cross shaped frame. Each of the rotors can be controlled independently, avoiding the need for a swashplate mechanism as in conventional helicopters. The propellers have fixed-pitch blades, producing a downward air flow. Quadcopters use the independent variation of the generated thrust of each rotor in order to achieve control.

Quadrotor control brings up an interesting and difficult problem. A quadcopter is an underactuated system, meaning it has six degrees of freedom (three

translational and three rotational) but only four independent actuators (its four propellers). This results in highly nonlinear dynamics, especially when taking into account the full array of complicated aerodynamic effects the system is subject to. Also, friction dampening reveals to be a small factor in preventing the motion of the quadrotor in the air, meaning that quadcopters have to provide their own damping to attain stability. In spite of this, thanks to the simple structure of the quadrotor and the ability to differentially control the speed of the rotors, it is easy to decouple its four inputs into four basic movements.

Lets us then define this four basic movements:

- **Throttle** ($U_1[N]$)

This command is implemented by varying all the propeller speeds by the same amount, and thus the thrust generated by each propeller. This command leads to a vertical force in the fixed-body frame. If the inertial frame and the body fixed frame coincide (the quadrotor is horizontal) the vertical direction of the force vector will be the same. Otherwise, an acceleration will be generated both in the horizontal and vertical plane, thus providing motion in the inertial $x - y$ plane.

Figure 3a shows a simplified schematic of the throttle command. Being Ω_r a constant rotor speed, and $\Delta\Omega_r$ a positive variable that represents an increment.

- **Roll** ($U_2[Nm]$)

This command is implemented by respectively increasing and decreasing (or vice versa) the right and left rotors speed by the same amount. This thrust differential generates a torque with respect to the x_B axis, generating a moment and allowing the quadrotor to turn. This command maintains the throttle constant since the overall thrust generated remains unchanged.

Figure 3b shows a simplified schematic of the roll command.

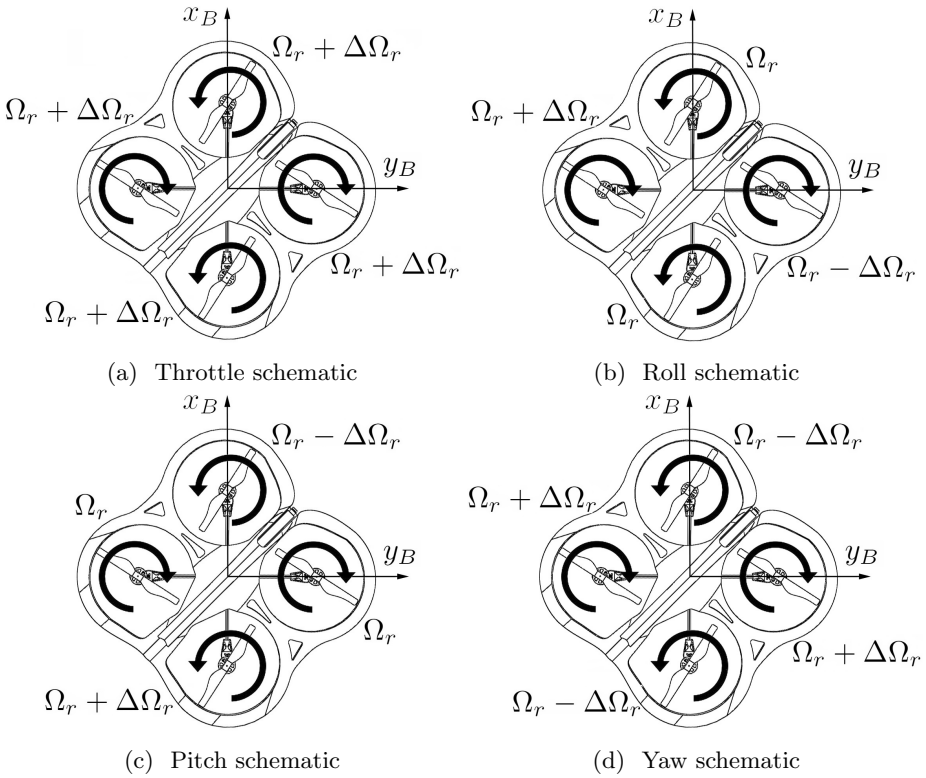
- **Pitch** ($U_3[Nm]$)

Similarly to the roll command, pitch is implemented by respectively increasing and decreasing (or vice versa) the front and rear rotors speed by the same amount. This thrust differential generates a torque with respect to the y_B axis, generating a moment and allowing the quadrotor to turn. This command maintains the throttle constant since the overall generated thrust remains unchanged. Pitch and Roll commands are the quadrotor means of changing the direction of the thrust vector produced by the throttle.

Figure 3c shows a simplified schematic of the roll command.

- **Yaw** ($U_3[Nm]$)

The objective of this command is to rotate the quadrotor around the z_b axis. This is achieved by respectively increasing and decreasing (or vice versa) the



(a) Throttle schematic (b) Roll schematic

(c) Pitch schematic (d) Yaw schematic

Figure 3: Throttle, Roll, Pitch and Yaw schematics

speed of the front-rear and right-left rotor couple by the same amount. This produces a torque with respect to the z_b axis. This moment is created thanks to the torque imbalance in the quadrotor, due to the different rotation direction of the propeller pairs. The front-rear propellers rotate counter-clockwise while the right-left pair rotates clockwise. As in the previous cases, the overall thrust generation remains constant

Figure 3d shows a simplified schematic of the roll command.

4.1.3 Newton-Euler model

There exist two typical methods to obtain the dynamic model of a rigid body system like a quadrotor. The Newton-Euler approach will be used since it is more simple and comprehensible, however other papers like [45] and [46] provide with the model derived from the Lagrangian method based on the concept of kinetic and potential energy.

In order to describe the motion of the quadrotor system two reference frames will be defined:

- Earth inertial frame (E)

- Fixed-body frame (B)

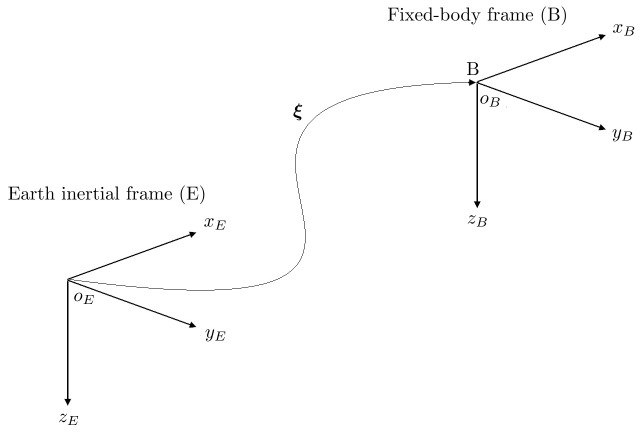


Figure 4: Earth and body coordinate system.

Concerning the inertial frame, E (o_E, x_E, y_E, z_E), a right handed NED (North-East-Down) coordinate system convention, typical of aviation applications, is used. In the NED convention x_E points towards the North, y_E points West and z_E points downwards into the earth's centre. The fixed-body frame, B (o_B, x_B, y_B, z_B), is fixed along the arms of the quadrotor as in Figure 5, where x_B points to the front of the quadrotor, y_B points to the right rotor and z_B points downwards while o_B is the axis origin and coincides with the crossing point of the arms of the structure. This frame also has a right hand reference.

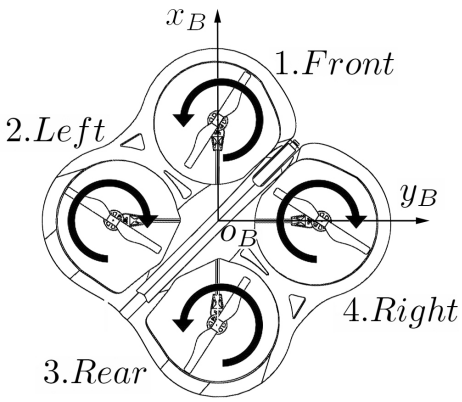


Figure 5: Quadcopter schematic

Let us then define the following workspace within this two frames:

$$\boldsymbol{V}^B = \begin{bmatrix} u & v & \omega \end{bmatrix}^T \tag{1}$$

$$\boldsymbol{\omega}^B = \begin{bmatrix} p & q & r \end{bmatrix}^T \tag{2}$$

Where the vector V^B $[m\ s^{-1}]$[1] represents the linear velocity vector of the body frame with respect to the inertial frame, being u, v and ω the velocities in the positive x, y and z directions respectively. Similarly, the vector $\boldsymbol{\omega}^B$ $[rad\ s^{-1}]$ represents the angular velocity of the quadrotor with respect to the inertial frame, being p, q and r the angular velocities corresponding to the roll, pitch and yaw movements.

Two different vectors for the linear and angular position will also be defined, $\boldsymbol{\Gamma}^E$ $[m]$ and $\boldsymbol{\Theta}^E$ $[rad]$:

$$\boldsymbol{\Gamma}^E = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{3}$$

$$\boldsymbol{\Theta}^E = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T \tag{4}$$

In equations 3 and 4, x, y and z represent the linear position of the body frame with respect to the earth frame. For the angular position the Euler angles convention was used, in $\boldsymbol{\Theta}^E$, ϕ stands for roll, θ for pitch and ψ for yaw of the body frame with respect to the inertial frame.

The rotation of the earth frame with respect to the body frame can now be expressed in a transform matrix in terms of the Euler angles, $\mathbf{R}_{\boldsymbol{\Theta}}(\phi, \theta, \psi)$ $[-]$. This rotation is comprised of 3 independent matrices, describing the individual rotation of the body about each of the inertial frame axis. The order of post-multiplication of these matrices will determine the order on which the rotations will be made.

The rotations about the x, y and z angles are given by the following matrices:

$$\mathbf{R}(\psi, z) = \begin{bmatrix} c(\psi) & -s(\psi) & 0 \\ s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

$$\mathbf{R}(\theta, y) = \begin{bmatrix} c(\theta) & 0 & s(\theta) \\ 0 & 1 & 0 \\ -s(\theta) & 0 & c(\theta) \end{bmatrix} \tag{6}$$

$$\mathbf{R}(\phi, x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi) & c(\phi) \end{bmatrix} \tag{7}$$

Where $c(\alpha) = cos(\alpha)$, $s(\alpha) = sin(\alpha)$ and $t(\alpha) = tan(\alpha)$. Then, post-multiplying this three matrices to obtain the rotation matrix from body frame to earth frame yields:

$$\mathbf{R}_{\boldsymbol{\Theta}} = \mathbf{R}(\psi, z) \cdot \mathbf{R}(\theta, y) \cdot \mathbf{R}(\phi, x) =$$

[1]The subscript B means that the referenced vector is defined in the body frame while the E subscript means the vector is defined in the inertial frame.

$$= \begin{bmatrix} c(\psi)c(\theta) & -s(\psi)c(\phi) + c(\psi)s(\theta)s(\phi) & s(\psi)s(\phi) + c(\psi)s(\theta)c(\phi) \\ s(\psi)c(\theta) & c(\psi)c(\theta) + s(\psi)s(\theta)s(\phi) & -c(\psi)s(\phi) + s(\psi)s(\theta)c(\phi) \\ -s(\theta) & c(\theta)s(\phi) & c(\theta)c(\phi) \end{bmatrix} \quad (8)$$

Using similar methodology the relationship between the angular velocities of the quadrotor (defined in the body frame) with respect to inertial frame can be determined. The transformation matrix $\mathbf{T_\Theta}$ is obtained by resolving the rate of the Euler angles $\dot{\mathbf{\Theta}}^E$ into the body frame:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \mathbf{R}(\phi, x)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \mathbf{R}(\phi, x)^{-1}\mathbf{R}(\theta, y)^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (9)$$

Then we have:

$$\mathbf{T}_\mathbf{\Theta}^{-1} = \mathbf{I_{3\times 3}} + \mathbf{R}(\phi, x)^{-1} + \mathbf{R}(\phi, x)^{-1}\mathbf{R}(\theta, y)^{-1} = \begin{bmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & c(\theta)s(\phi) \\ 0 & -s(\phi) & c(\theta)c(\phi) \end{bmatrix} \quad (10)$$

with $\mathbf{I_{3\times 3}}$ being a three by three identity matrix, then:

$$\mathbf{T_\Theta} = \begin{bmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & -s(\phi)/c(\theta) & c(\phi)/c(\theta) \end{bmatrix} \quad (11)$$

Two vectors can be defined to give a generalized overview of the position and velocity of the quadrotor in the space:

$$\boldsymbol{\xi} = \begin{bmatrix} \mathbf{\Gamma}^E & \mathbf{\Theta}^E \end{bmatrix}^T = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}^T \quad (12)$$

$$\boldsymbol{\nu} = \begin{bmatrix} \mathbf{V}^B & \boldsymbol{\omega}^B \end{bmatrix}^T = \begin{bmatrix} u & v & w & p & q & r \end{bmatrix}^T \quad (13)$$

Where $\boldsymbol{\xi}$ [+] represents the generalized position of the body with in terms of the earth frame. and $\boldsymbol{\nu}$ [+] the generalized velocity in terms of the body frame.

Using equations 8 and 11 to switch between the different frames of reference it can be shown that the following relationships stand true:

$$\mathbf{V}^E = \dot{\mathbf{\Gamma}}^E = \mathbf{R}_\Theta \mathbf{V}^B \quad (14)$$

$$\boldsymbol{\omega}^B = \mathbf{T}_\mathbf{\Theta}^{-1}\dot{\mathbf{\Theta}}^E \quad (15)$$

$$\dot{\mathbf{\Theta}}^E = \mathbf{T}_\mathbf{\Theta}\boldsymbol{\omega}^E \quad (16)$$

And the same can be done with the generalized position of the body $\boldsymbol{\xi}$:

$$\boldsymbol{\xi} = \mathbf{G}_{\boldsymbol{\Theta}}\, \boldsymbol{\nu} \tag{17}$$

Where:

$$\mathbf{G}_{\boldsymbol{\Theta}} = \begin{bmatrix} \mathbf{R}_{\boldsymbol{\Theta}} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{T}_{\boldsymbol{\Theta}} \end{bmatrix} \quad \in \mathbb{R}^{6\times 6} \tag{18}$$

Now that a suitable kinematic workspace has been defined the dynamic description of the system can be developed. In order to simplify this task the following assumptions concerning the formulation of the equations have been made:

1. The body-fixed frame axes coincide with the principal axes of inertia of the body
2. The moments of inertia are constant.
3. The body-fixed frame origin o_B is coincident with the centre of mass.
4. Body symmetry with respect to the centre of mass is assumed.

This assumptions reduce notoriously the complexity of the model. The centre of mass coincidence with o_B implies that the actuators action is symmetrical thus simplifying the need of complex centre of mass shifts in rigid body calculations. Besides this, the first assumption implies that the inertia matrix $\boldsymbol{I}$ is diagonal, simplifying again the equations.

The rigid body dynamic equations of the quadrotor come from the application of Newton's second law, according to equation 19:

$$\begin{aligned} m\ddot{\boldsymbol{\Gamma}}^E &= \mathbf{F}^E \\ m\dot{\widehat{\mathbf{R}_{\boldsymbol{\Theta}}\mathbf{V}^B}} &= \mathbf{R}_{\boldsymbol{\Theta}}\mathbf{F}^B \\ m(\mathbf{R}_{\boldsymbol{\Theta}}\dot{\mathbf{V}}^B + \dot{\mathbf{R}}_{\boldsymbol{\Theta}}\mathbf{V}^B) &= \mathbf{R}_{\boldsymbol{\Theta}}\mathbf{F}^B \\ m\mathbf{R}_{\boldsymbol{\Theta}}(\dot{\mathbf{V}}^B + \boldsymbol{\omega}^B \times \mathbf{V}^B) &= \mathbf{R}_{\boldsymbol{\Theta}}\mathbf{F}^B \\ m(\dot{\mathbf{V}}^B + \boldsymbol{\omega}^B \times \mathbf{V}^B) &= \mathbf{F}^B \end{aligned} \tag{19}$$

Where $m\ [Kg]$ is the mass of the quadrotor, $\ddot{\boldsymbol{\Gamma}}^E[m\ s^{-2}]$ is the second derivative of the linear position of the body frame with respect to the earth frame, $\mathbf{F}^E[N]$ and $\mathbf{F}^B[N]$ are the forces vector with respect to the earth and body frame, $\mathbf{R}_{\boldsymbol{\Theta}}$ is the rotation matrix and $\mathbf{V}^B$ and $\boldsymbol{\omega}^B$ are the body frame linear and angular speeds of the quadrotor expressed in the body frame.

Then dynamic angular components of the body motion are also obtained from Newton's second law in a similar manner. Equation 20 shows the shortened development of this expression:

$$\boldsymbol{I}\ddot{\boldsymbol{\Theta}}^{E} = \boldsymbol{\tau}^{E}$$

$$\vdots$$

$$\boldsymbol{I}\dot{\boldsymbol{\omega}}^{B} + \boldsymbol{\omega}^{B} \times (\boldsymbol{I}\boldsymbol{\omega}^{B}) = \boldsymbol{\tau}^{B} \tag{20}$$

Where $\boldsymbol{I}$ $[N\ m\ s^2]$ is the inertia matrix of the body (with respect to the body frame), $\ddot{\boldsymbol{\Theta}}^{E}[rad\ s^{-2}]$ is the second derivative of the angular position of the quadrotor with respect to the earth frame, $\boldsymbol{\tau}^{E}[N\ m]$ and $\boldsymbol{\tau}^{B}[N\ m]$ are the torques vectors with respect to the earth and body frame.

Equations 19 and 20 are the generic expressions of Newton's second law in a three dimensional space for a 6-DOF rigid body. Equation 21 expresses this two equations in matrix form.

$$\begin{bmatrix} m\boldsymbol{I}_{\mathbf{3\times 3}} & \mathbf{0_{3\times 3}} \\ \mathbf{0_{3\times 3}} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{V}}^{B} \\ \dot{\boldsymbol{\omega}}^{B} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}^{B} \times (m\mathbf{V}^{B}) \\ \boldsymbol{\omega}^{B} \times (\boldsymbol{I}\boldsymbol{\omega}^{B}) \end{bmatrix} = \begin{bmatrix} \mathbf{F}^{B} \\ \boldsymbol{\tau}^{B} \end{bmatrix} \tag{21}$$

In order to apply this model to describe the quadrotor dynamics the forces and torques applied to the body must be defined. A generalized force and torque vector $\boldsymbol{\Lambda}[+]$ can be defined, described in equation 22, that will help describing the different components of this forces.

$$\boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}^{B} \\ \boldsymbol{\tau}^{B} \end{bmatrix} = \begin{bmatrix} F_x & F_y & F_z & \tau_x & \tau_y & \tau_z \end{bmatrix}^{T} \tag{22}$$

The forces and torques acting on the quadrotor $\boldsymbol{\Lambda}$ can be divided in several contributions depending of their source. This forces are modelled directly on the body frame since they are easier to implement later on. $\boldsymbol{\Lambda}$ can be described as a sum of all the forces at play in the quadrotor. Rewriting the expression for $\boldsymbol{\Lambda}$ as in equation 23 divides it into its different components.

$$\boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{F}^{B} \\ \boldsymbol{\tau}^{B} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{G}^{B} \\ \mathbf{0_{3\times 1}} \end{bmatrix} + \mathbf{U}^{B} + \begin{bmatrix} \mathbf{0_{3\times 1}} \\ \mathbf{G}_{a}^{B} \end{bmatrix} + \mathbf{F}_{ext}^{B} + \mathbf{F}_{other}^{B} \tag{23}$$

Where $\mathbf{F}_{G}^{B}[N]$ is the gravitational contribution, $\mathbf{U}^{B}[+]$ is the actuator contribution, $\mathbf{G}_{a}^{B}[N\ m]$ is the gyroscopic contribution, $\mathbf{F}_{ext}^{B}[+]$ is the external forces vector and $\mathbf{F}_{other}^{B}[+]$ contains the contribution of other aerodynamic effects not considered in the previous components. Each of these components is further described bellow:

- **Gravitational force component**

  This component comes from the effect of gravity on the quadrotor. This component only affects the linear equations of the model. This contribution can be modelled through the following expression:

$$\mathbf{F}_{G}^{B}(\boldsymbol{\Theta}) = \mathbf{R}_{\boldsymbol{\Theta}}^{-1}\mathbf{F}_{G}^{E} = \mathbf{R}_{\boldsymbol{\Theta}}^{T} \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} = \begin{bmatrix} -m\ g\ s(\theta) \\ m\ g\ c(\theta)s(\phi) \\ m\ g\ c(\theta)s(\phi) \end{bmatrix} \tag{24}$$

Where $\mathbf{F}_G^B$ is the force vector due to the gravitational contribution expressed in the body frame. As it can be seen in equation 24, the force vector points always downwards in the inertial frame and is translated into the body frame by means of the rotation matrix previously calculated.

- **Input contribution**

  We can proceed now to model the actuators contribution to the model. The given definition in section 4.1.2 of the four basic basic movements of the quadrotor will be used, namely Throttle (U_1), Roll (U_2), Pitch (U_3) and Yaw (U_4). Using the same notation, they can be defined as:

$$U_1 = -\sum_{i=1}^{4} |T_i| = -c_T(\sum_{i=1}^{4} \Omega_i^2) \tag{25}$$

$$U_2 = c_T\ l(\Omega_2^2 - \Omega_4^2) \tag{26}$$

$$U_3 = c_T\ l(\Omega_1 v - \Omega_3^2) \tag{27}$$

$$U_4 = c_D(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \tag{28}$$

  Where T_i and Ω_i are the generated thrust and speeds of each rotor respectively, with the same notation as in Figure 5, c_T $[N\ s^2]$ and c_D $[N\ m\ s^2]$ are the thrust and drag coefficients and l $[m]$ is the distance between the centre of the quadrotor and the centre of the propeller. Equation 29 gives a lumped vector $\mathbf{U}^B[+]$ containing the four components generated by the actuators. A more in depth coverage of the modelling of thrust and drag forces is given in Appendix A

$$\mathbf{U}^B = \begin{bmatrix} 0 \\ 0 \\ U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -c_T(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\ c_T\ l(\Omega_2^2 - \Omega_4^2) \\ c_T\ l(\Omega_1^2 - \Omega_3^2) \\ c_D(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \end{bmatrix} \tag{29}$$

- **Gyroscopic effects due to propeller rotation**

  Propeller rotation also produces a gyroscopic effect besides the ones derived from the rigid body mechanics of the airframe. This gyroscopic torque is produced by the combination of the rotation of the airframe and the four rotors. The expression of this gyroscopic torque is given by the following expression [47] :

$$G_a^B = \sum_{i=1}^{4} \boldsymbol{I}_p(\boldsymbol{\omega}^B \times e_z)(-1)^{i+1}\Omega_i \qquad \in \mathbb{R}^{3\times 1} \tag{30}$$

  Where $\boldsymbol{I}_p$ $[N\ m\ s^2]$ is the moment of inertia of the propeller, e_z is a unitary vector in the z_E direction and Ω_i the speed of each rotor.

From the development of the cross product $\boldsymbol{\omega}^B \times e_z$ in equation 31 it can be seen that this torque is directly related to the angular rate of the body frame. From this expression we can also derive that no gyroscopic torque occurs around yawing axis since the yaw torque and the propeller axes are already parallel. Gyroscopic torques in quadrotors are more deeply analysed in [48] , [49] and [47].

$$\boldsymbol{\omega}^B \times e_z = \begin{bmatrix} q \\ -p \\ 0 \end{bmatrix} \tag{31}$$

- **External forces**

  External forces may be applied to the quadrotor while in flight. This force can be used to model wind or other forces besides those of the actuators. This is an important contribution since wind resistance, for example, provides dampening in both linear and angular motions of the body.

- **Other aerodynamic forces**

  While the dominant aerodynamics are governed by the actuator forces, other aerodynamic forces, like the aforementioned gyroscopic torque, are produced within the frame of the quadrotor or the propellers. This effects can be neglected since they only cause minor disturbances and do not influence noticeably the control problem to be solved in this work.

  Some of the most important ones are blade flapping and induced drag that are covered in Appendix A. A more in depth analysis can be found in the work of [9], where they briefly but concisely describe this effects.

Now that all the necessary components of the equations of motion have been defined equation 21 can be rewritten in matrix form:

$$\boldsymbol{M_{RB}\dot{\nu} + C_{RB}(\nu)\nu = \Lambda} \tag{32}$$

Where:

$$\boldsymbol{M_{RB}} = \begin{bmatrix} \boldsymbol{mI_{3\times 3}} & \boldsymbol{0_{3\times 3}} \\ \boldsymbol{0_{3\times 3}} & \boldsymbol{I} \end{bmatrix} \tag{33}$$

$$\boldsymbol{C_{RB}} = \begin{bmatrix} \boldsymbol{0_{3\times 3}} & \boldsymbol{-m\mathbf{S}(\mathbf{V}^B)} \\ \boldsymbol{0_{3\times 3}} & \boldsymbol{-\mathbf{S}(I\omega^B)} \end{bmatrix} \tag{34}$$

As it can be seen, $\boldsymbol{M_{RB}}$ is expressed as a diagonal matrix thanks to the aforementioned assumptions made of the quadcopter model. In the expression for $\boldsymbol{C_{RB}}$ the notation $\mathbf{S}(k)[+]$ stands for the skew-symmetric matrix operator. This operator is defined by equation 35:

$$\boldsymbol{S(k) = -S^T(k)} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{-k_3} & \boldsymbol{k_1} \\ \boldsymbol{k_3} & \boldsymbol{0} & \boldsymbol{-k_1} \\ \boldsymbol{-k_2} & \boldsymbol{k_1} & \boldsymbol{0} \end{bmatrix} \qquad \boldsymbol{k} = \begin{bmatrix} \boldsymbol{k_1} \\ \boldsymbol{k_2} \\ \boldsymbol{k_3} \end{bmatrix} \tag{35}$$

By rearranging equation 33 the derivative of the generalized velocity of the body $\dot{\boldsymbol{\nu}}^B$ can be isolated to obtain the following expression:

$$\dot{\boldsymbol{\nu}} = \boldsymbol{M_B}^{-1}(-\boldsymbol{C_B}(\boldsymbol{\nu})\boldsymbol{\nu} + \begin{bmatrix} \mathbf{F}_{\boldsymbol{G}}^{\boldsymbol{B}} \\ \mathbf{0_{3\times 1}} \end{bmatrix} + \mathbf{U}^{\boldsymbol{B}} + \begin{bmatrix} \mathbf{0_{3\times 1}} \\ \mathbf{G}_{\boldsymbol{a}}^{\boldsymbol{B}} \end{bmatrix} + \mathbf{F}_{\boldsymbol{ext}}^{\boldsymbol{B}} + \mathbf{F}_{\boldsymbol{other}}^{\boldsymbol{B}}) \quad (36)$$

The system of equations form of the above matrix equation is shown in equation 37

$$\begin{cases} \dot{u} &= (v\ r - w\ q) - g\ c(\theta) + \dfrac{F_{x_x}}{m} \\ \dot{v} &= (w\ p - u\ r) + g\ c(\theta)s(\phi) + \dfrac{F_{y_x}}{m} \\ \dot{w} &= (u\ q - v\ p) + g\ c(\theta)s(\phi) + \dfrac{U_1 + F_{z_x}}{m} \\ \dot{p} &= \dfrac{I_{YY} - I_{ZZ}}{I_{XX}} q\ r + \dfrac{I_p}{I_{XX}} q\ \Omega + \dfrac{U_2 + \tau_{x_x}}{I_{XX}} \\ \dot{q} &= \dfrac{I_{ZZ} - I_{XX}}{I_{YY}} p\ r + \dfrac{I_p}{I_{YY}} p\ \Omega + \dfrac{U_3 + \tau_{y_x}}{I_{YY}} \\ \dot{r} &= \dfrac{I_{XX} - I_{YY}}{I_{ZZ}} p\ q + \dfrac{U_4 + \tau_{z_x}}{I_{ZZ}} \end{cases} \quad (37)$$

Where $F_{\cdot_x}$ and $\tau_{\cdot_x}$ denotes the sum of the force and torque components of the external and other aerodynamic forces and:

$$\Omega = \sum_{i=1}^{4} \Omega_i (-1)^{i+1} \quad (38)$$

4.2 Unmanned Surface Vehicle: Strider V1.0

This chapter will present the Unmanned Surface Vehicle to be simulated, the Strider V1.0. The first section of this chapter (4.2.1) will present the platform to be used in the project, the second one (4.2.2) delves into the basic concepts of modelling of surface vehicles and finally (4.2.3) deals with the Newton-Euler model of the platform.

4.2.1 The Strider V1.0

The Strider V1.0, seen in Figure 6, is a catamaran based small surface vessel built by David Borreguero. The design and development process is available in [3]. The vessel features a landing platform for the quadrotor with an electronic box beneath it to accommodate all the required control systems for the autonomous operation. Two parallel hulls of the same size are attached to the landing platform by a PVC tubing structure. The Strider gets his thrust from a $40mm$ propeller and is controlled by a single central rudder. Figure 7 shows the back of the platform with the opened electronic box along with the propeller and the rudder. The ultrasonic sensors used for obstacle avoidance can also be seen attached to the platform.

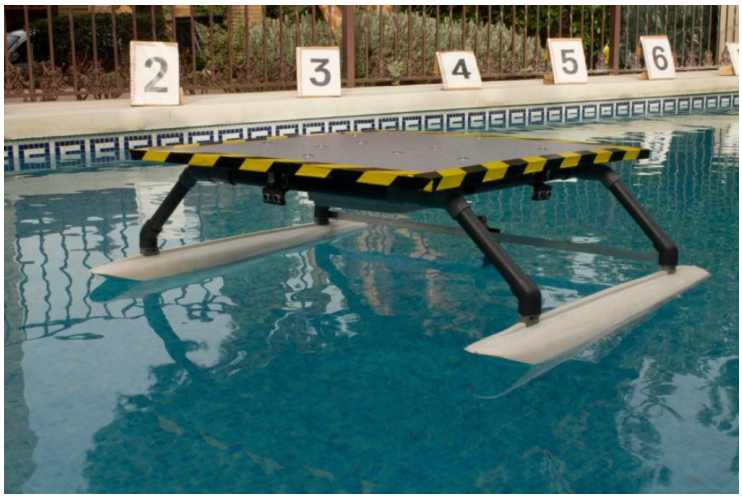


Figure 6: The Strider V1.0. From [3].

Table 5 contains some of the most relevant parameters to this work of the vessel. The complete feature set of the vessel can be found in [3]. The control of the vessel is achieved using an Arduino Mega microcontroller that is fed up by the sensory equipment installed in the platform. This information can be either transferred to be processed by an outside control station via bluetooth or directly processed by the microcontroller introducing less delay in the control loop. Sensory wise, the vessel carries four ultrasonic sensors for obstacle avoidance duties and an IMU (Inertial Measurement Unit) to keep track of the vessel's position and attitude.

	Parameter	Value
Main platform	Mass	420 gr
	Length	85 cm
	Width	75 cm
	Thickness	2 cm
Rudder	Area	64 cm^2
	Height	9 cm
	Maximum rudder speed	353 $/s$
Hull	Mass	652 gr
	Length	91 cm
	Width	12.5 cm
	Average Height	7 cm
Propeller	Diameter	40 mm
Total structure	Mass	4.310 kg

Table 5: Strider V1.0 features.

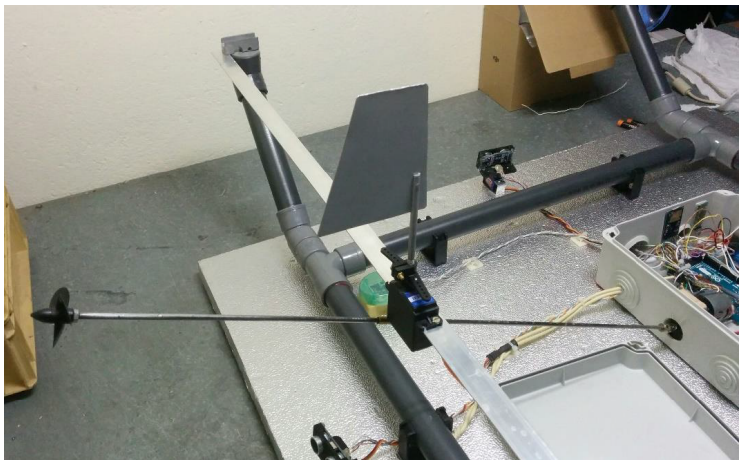


Figure 7: Back of the landing platform of the Strider V1.0. From [3].

4.2.2 Surface Vessel modelling

Many different approaches to surface vessel motion modelling can be found in the literature such as [50], [51], [52] and [53]. Chapters 2 and 5 of Fossens's *Guidance Control of Ocean Vehicles* [54] provides a compilation of a wide array of different models for the description of the motion of a vessel. Generally these models are decomposed into speed (surge), steering (sway and yaw) and the rest of degrees of freedom of the vessel, but take different modelling approaches. Another important difference to be noted in these models is whether the presented models are linear or non-linear.

When considering steering models, Nomoto's model [51] is amid the most simpler and popular linear steering models. This model provides with a 2nd order transfer function that relates the yaw angle ψ with the rudder angle δ and it has been successfully used in many modelling and control design approaches of surface vehicles as in [55] and [56]. This model is widely used in ship autopilot design due to its simplicity and accuracy although it does not accommodate slipping motion.

Norbin's model [50] provides empirical representations of the hydrodynamic forces in the $x - y$ plane and yaw moments in order to develop a nonlinear expression for the motion of the vessel. This model also includes an approach to effectively model rudder angle and generated thrust. Blanke's model [57] sets a simplified form of Norbin's work retaining only the most important terms for steering and propulsion. Both models are based on the assumption that only horizontal motion of the ship is allowed and only surge, sway and yaw motions are taken into account.

Some models, both linear and non-linear, also consider rolling besides the horizontal motion assumed in Norbin's or Blanke's models. Christensen and Blanke present in [58] how the steering and roll set of equations can be approximated in a nonlinear manner and then show a linearised state-space model of

this same set of equations. Other works on coupled steering and roll motion can be found in [59] and [60].

All these different models can be divided into two grand groups depending on the approach they are based on. We can differentiate between Manoeuvring Theory and Seakeeping Theory. A more in depth analysis of these models classification can be found in Appendix B.

4.2.3 Newton-Euler model

Generally most of the works dealing with ship modelling use a Newtonian approach. Nonetheless, some works such as [61], [62] and [63] do present with the Lagrange derived equations of motion of the vessel. In this work the Newtonian approach was selected to obtain the dynamic model for the Strider V.0.

For the dynamic modelling of the Strider V1.0 a similar rigid-body mechanical model to the one used in section 4.1.3 is adopted. An inertial and a body fixed frame will be defined to create a suitable workspace for the model. The SNAME (Society of Naval Architects and Marine Engineers) provides with a standard notation and sign convention for the description of the motion of ships shown in Figure 8.

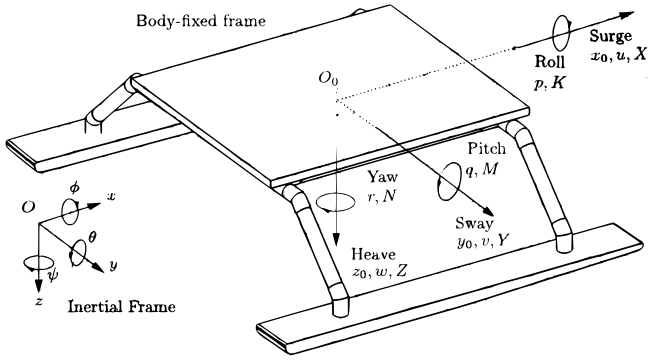


Figure 8: Standard notation and sign conventions for ship motion description on the Strider V1.0

As seen in the figure, both frames show a right hand rule convention with z downwards. The inertial frame is the North-East-Down frame (NED) reference frame attached to the Earth. The positive x axis points towards the North, the positive y axis towards the East, and the positive z axis towards the centre of the Earth.

On the other hand, a body frame fixed to the vessel is considered. Generally for surface ships the centre position for the body fixed frame is located in such a way that it gives hull symmetry about the $x_0 - z_0$ plane and approximate symmetry about the $y_0 - z_0$ plane while the origin of the z_0 axis is set on the calm water surface. In our particular case, due to the symmetry and small size of platform, the hydrodynamic forces and moments acting on the ship can be

easily described when the centre of the body-frame is coincident with the centre of gravity of the vessel. Thus, the position of O_0 will be assumed to be located at the centre of mass of the Strider V1.0

Angular motion is defined in terms of the Euler angles being $\begin{bmatrix}\phi & \theta & \psi\end{bmatrix}[rad]$ the roll, pitch and yaw angles respectively as shown in Figure 8. We will also define $\begin{bmatrix}p & q & r\end{bmatrix}[rad\ s^{-1}]$ and $\begin{bmatrix}K & N & M\end{bmatrix}[N\ m]$ as the angular velocities and the torques about the x, y and z directions with respect to the body frame.

Concerning linear motion $\begin{bmatrix}x & y & z\end{bmatrix}[m]$ stands for the linear position of the body fixed frame with respect to the inertial frame. While $\begin{bmatrix}u & v & w\end{bmatrix}[m\ s^{-1}]$ stands for the linear velocity of the vessel and $\begin{bmatrix}X & Y & Z\end{bmatrix}[N]$ for the forces and torques applied to the body, both referenced in the body fixed frame. Motions on the $x-y-z$ direction are usually called surge, sway and heave respectively.

With this in mind the following notation will be used:

$$\boldsymbol{\eta}^E = \begin{bmatrix}x & y & z & \phi & \theta & \psi\end{bmatrix}^T \tag{39}$$

$$\boldsymbol{\nu}^B = \begin{bmatrix}u & v & w & p & q & r\end{bmatrix}^T \tag{40}$$

$$\boldsymbol{\tau}^B = \begin{bmatrix}X & Y & Z & K & M & N\end{bmatrix}^T \tag{41}$$

Where $\boldsymbol{\eta}^E[+]$ is the linear and angular position with respect to the inertial frame, $\boldsymbol{\nu}^B[+]$ the linear and angular velocity of the ship with respect to the body frame and $\boldsymbol{\tau}^B[+]$ represents the forces and torques applied to the body in terms of the body fixed frame.

The dynamic motion of the vessel can be described by the expression in equation 42 where $\boldsymbol{M_{RB}}$ is the mass and inertia matrix, $\boldsymbol{C_{RB}(\nu)\nu}$ accounts for the coriolis and centripetal forces and moments derived from the rigid body dynamics and $\boldsymbol{\tau}^B$ is the forces and moments vector defined above. This expression is analogue to the newtonian expression of rigid body dynamics made in equation 32.

$$\boldsymbol{M_{RB}\dot{\nu} = \tau^B(\dot{\nu}, \nu, \eta) - C_{RB}(\nu)\nu} \tag{42}$$

As can be seen in the above equation, the force and torque vector $\boldsymbol{\tau}^B$ is dependent on both the acceleration and velocity of the ship and its linear position and attitude. This vector can be divided into different terms according the phenomena it is generated from:

$$\boldsymbol{\tau}^B = \boldsymbol{\tau}_{hyd} + \boldsymbol{\tau}_{cs} + \boldsymbol{\tau}_{ext} \tag{43}$$

Where:

- $\boldsymbol{\tau}_{hyd}$ are the forces and moments generated by the movement of the hull in the water. This includes hydrodynamic added mass potential damping, viscous damping and hydrostatic forces.

- $\boldsymbol{\tau}_{cs}$ are the forces generated by control surfaces (rudder, fins, etc.) and propulsion forces.

• $\boldsymbol{\tau}_{ext}$ are the forces and moments generated by exogenous contributions. They usually are environmental forces and can be usually separated into three categories: wind induced forces, wave induced forces and current induced forces.

In this work Manoeuvring Theory is adopted to model our simulation. Manoeuvring Theory involves the study of the ship's movement at a constant or slowly varying positive speed. A three degrees of freedom approach is commonly considered, where only surge, sway and yaw are analysed. Restricted, calm water and still waves are assumed. As treated in [64], the assumption of calm water or zero frequency wave excitation implies that the potential added mass and damping terms can be represented by using constant potential coefficients (hydrodynamic derivatives) to model the forces and torques at play.

The validity of this three degrees of freedom approach to control design is based on the assumption that the surge, sway and yaw natural period of the vessel, when controlled by a PD control loop, will be in the range of $100 - 150\ s$. This results in close to zero natural frequency values of these parameters which validates the assumption of constant hydrodynamic derivatives [64].

Since heave, roll and pitch motions tend to have much higher natural frequencies they violate this zero frequency assumption. This means that their hydrodynamic constants would have to be evaluated at frequencies different and higher than zero. Because of this, the zero frequency hydrodynamic derivatives of the heave, roll and pitch motions are usually neglected in manoeuvring models.

The made assumptions can be summarized in the following list:

1. The body-fixed frame axes coincide with the principal axes of inertia of the body
2. The moments of inertia are constant.
3. Body symmetry with respect to the centre of mass is assumed.
4. The body-fixed frame origin O_0 is coincident with the centre of mass.
5. Restricted, calm and still water bodies is assumed. This implies that no currents or waves affect the motion of the ship.
6. Heave, roll and pitch motions are neglected due to a zero frequency wave excitation assumption.
7. Surge motion is decoupled from sway and yaw motion due to the symmetry of the vessel hulls.
8. Added mass effects on the hulls are neglected since only steady motion will be considered.

With this in mind the 3DOF dynamic model can be modelled, taking from equation 42, as:

$$\boldsymbol{M}^*_{\boldsymbol{RB}}\dot{\boldsymbol{\nu}}^* = \boldsymbol{\tau}^*(\dot{\boldsymbol{\nu}}^*, \boldsymbol{\nu}^*, \boldsymbol{\eta}^*) - \boldsymbol{C}^*_{\boldsymbol{RB}}(\boldsymbol{\nu}^*)\boldsymbol{\nu}^*$$

$$\begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} X \\ Y \\ N \end{bmatrix} - \begin{bmatrix} 0 & -mr & 0 \\ mr & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} \tag{44}$$

Where the subscript * stands for the reduced 3DOF version of the terms in equation 42.

To finish this chapter a description of the forces and torques that act on the vessel is given. This description is divided in three parts, covering each of the terms of equation 43.

Hydrodynamic forces.

We will now deal with the modelling of expressions for the hydrodynamic forces and moments. A more in depth analysis of hydrodynamic forces is given in Appendix B.

Based on the assumptions made concerning the interaction between the vessel's hull and the water, namely the zero frequency wave and the calm water and the steady motion assumption, it can expected that the only source of hydrodynamic effects is produced just by the motion of the body in a viscous fluid.

The analysis of the motion of vessels in water bodies yields a variety of complex interacting mechanisms that generate hydrodynamic resistance, understood the as the dynamic response to the motion of the hull in a viscous fluid. In Lewis' *"Principles of Naval Architecture"* [65] this effects are analysed and results concerning the hydrodynamic resistance in calm water bodies presented. For a calm water body, the resistance experienced by the vessel movement can be assumed to comprised of these three components:

- Frictional resistance, due to the motion of the ship's hull in viscous fluid.
- Wave making resistance, created by the energy loss due to the generated waves produced by the hull's motion in water.
- Eddy resistance, produced by the energy carried away by eddies generated from the hull.

Along with this effects we can add the contribution made by Vortex Sheedding damping and lift and drag forces produced by the hull's displacement in water, much like airfoils in aerial vehicles.

Other effects not considered in this chapter, since they lay outside our assumptions, like restoring forces (Buoyancy and weight), wave induced forces or added mass effects are covered in Appendix B.

The analytical expression for the hydrodynamic forces is expressed in terms of the hydrodynamic coefficients. In this work the modelling approach made by

Davidson and Schiff [52] will be used. This approach gives a linear approximation of the hydrodynamic forces based on the assumptions made in this work. As said before, surge motion can be assumed to be decoupled from sway and yaw, thus the expression for the transversal force and the yaw moment given by this model can be approximated as:

$$Y = Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} + Y_v v + Y_r r + Y_\delta \delta_R \tag{45}$$

$$N = N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} + N_v v + N_r r + N_\delta \delta_R \tag{46}$$

Where the constant coefficients are:

$$Y_{\dot{v}} = \frac{\partial Y}{\partial \dot{v}} \qquad Y_{\dot{r}} = \frac{\partial Y}{\partial \dot{r}} \qquad \cdots \tag{47}$$

This coefficients are the hydrodynamic derivatives, and they are an expression of the force produced in a direction due to the effects of linear and angular motion. For example, the first term in equation 47 is interpreted as the force produced in the positive y_0 direction due to the acceleration $\dot{v}$.

Surge force can be expressed in terms of two main contributions. The thrust provided by the propeller of the vessel and the hydrodynamic drag produced by the motion of the hull in the water. The expression for the thrust force will be given in later, when propulsion forces are covered.

$$X = X_{\dot{u}}\dot{u} + X(u) + T \tag{48}$$

Where $X_{\dot{u}}\dot{u}$ is the hydrodynamic derivative associated with added mass effects, $X(u)$ is the hydrodynamic resistance (which is a function of the forward speed) and T is the thrust generated by the propeller.

Control surfaces, the rudder.

Rudders are the usual control device used in marine vessels. Other steering and stabilizing methods range from differential thrust from more than one propulsive device to create yaw moments or by changing the direction of the thrust like outboard engines do in small boats.

Rudder generated forces are dependent on the geometry of the rudder, its position and orientation with respect to the vessel hulls and the speed of the flow. This force acts in a single point of the rudder, called centre of pressure $\boldsymbol{CP^B} = \begin{bmatrix} x_{cp} & y_{cp} & z_{cp} \end{bmatrix}$. This force is normal to the rudder plane. The value of the resulting force on the rudder is expressed as [5]:

$$F_R = \begin{cases} \frac{1}{2}\rho C_L A_r V_{av}^2 sin(\frac{\pi}{2}\frac{\delta_{attack}}{\delta_{stall}}) & if|\delta_{attack}| < \delta_{stall} \\ \frac{1}{2}\rho C_L A_r V_{av}^2 sign(\delta_{attack}) & if|\delta_{attack}| \geqslant \delta_{stall} \end{cases} \tag{49}$$

Where ρ $[Kg\ m^-3]$ is the density of the water, C_L $[-]$ is the lift coefficient of the rudder, A_r $[m^2]$ is the rudder area, V_{av} $[m\ s^{-2}]$ is the average flow passing the rudder and δ_{stall} $[rad]$ is the stall angle of the rudder. The function *sign(k)* gives back the sign of k. The angle of attack δ_{attack} $[rad]$ is the angle between

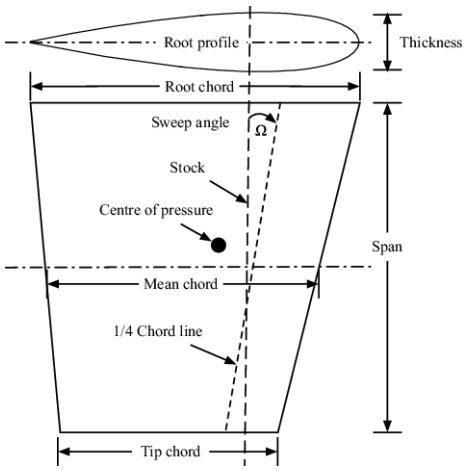


Figure 9: Rudder dimensions notation. From [4].

the plane of the rudder and the direction of the flow passing by the rudder. This angle is a function of the rudder angle δ and the sway, surge and yaw rates:

$$\begin{aligned} \delta_{attack} &= \delta - \delta_{flow} \\ \delta_{attack} &= \delta - arctan\left(\frac{v + (x_{CP} - x_{CG})}{u}\right) \end{aligned} \tag{50}$$

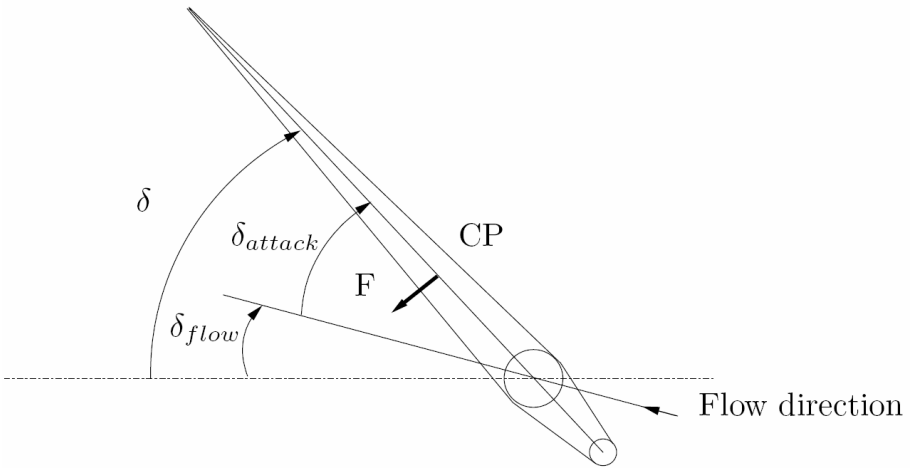


Figure 10: Rudder angle convention and notation. From [5].

Figure 10 shows an schematic of the rudder and the notation used in equation 49 to describe the rudder force. This force can be easily decomposed into its orthogonal components parallel to the x and y directions in the body fixed reference frame:

$$
\begin{aligned}
X_{rudder} &= -F_R(u, V_{av}, v, r, \delta) sin(\delta) \\
Y_{rudder} &= F_R(u, V_{av}, v, r, \delta) cos(\delta) \\
Z_{rudder} &= 0
\end{aligned} \tag{51}
$$

The moment produced in the vessel by this rudder force is calculated right away by multiplying the distance between the centre of mass of the vessel $\boldsymbol{CG}$ and the centre of pressure $\boldsymbol{CP^B}$. It will assumed that the centre of pressure and centre of gravity are in the same plane, along the centreline of the vessel.

$$
N_{rudder} = (\boldsymbol{CP^B} - \boldsymbol{CG}) F_R(u, V_{av}, v, r, \delta) cos(\delta) \tag{52}
$$

Equation 52 shows the expression for the yaw moment generated by the rudder actuation. Pitch moment due to rudder forces is not produced since no vertical forces are generated by the rudder. Also, roll moments are neglected due to the assumptions made in the model.

In cases where the propeller is located in front of the rudder (not in the case of this work), the average flow speed V_{av} is highly influenced by the propeller, [66] proposed that this influence could be modelled as:

$$
V_{av}^2 = V_a^2 + C_T T \qquad C_T = \frac{6.4}{\pi \rho h D_p} \tag{53}
$$

where V_a is the speed of advance, h is the rudder span and D_p the propeller diameter.

Propulsion.

Concerning propulsion devices, propellers are the standard thruster found in most surface vehicles, while more advanced propulsion systems like Voith propellers, pump-jets, impellers, etc. are less frequently found. Thrust generation is needed in order to overcome the drag forces produced by the hydrodynamic forces produced by the hull's motion in water and to keep the ship in motion.

A quick overview of thrust generation and an expression for this thrust is given. More complex and thorough analysis of propulsion forces in naval architecture can be found in [65].

In this work the generated torque by the propeller will not be considered since its contribution to surge, sway and yaw is negligible as a result of its direction. The thrust provided by the propeller is a function of its geometry, the spinning rate of the propeller, the flow speed and the fluid properties in which the propeller is submerged. Dimensional analysis of this problem yields the following expression for the generated thrust [67]:

$$
C_F = \frac{T}{\rho n^2 D^4} \qquad then, \tag{54}
$$

$$
T = C_F \rho n^2 D^4 \tag{55}
$$

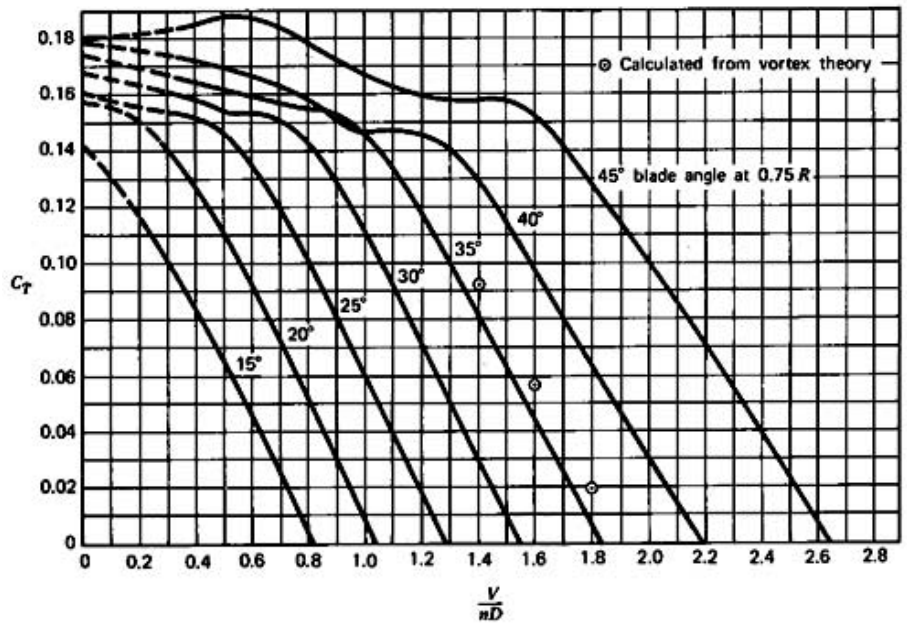


Figure 11: Typical thrust coefficient curves as a function of the advance ratio and pitch angle. From [6].

Where T $[N]$ is the generated thrust, ρ $[Kg\ m^-3]$ is the density of water, n $[Hz]$ the propeller speed in revolutions per second and D $[m]$ is the diameter of the propeller. Typical values for the thrust coefficient can be obtained from the works of McCormick [6]. Figure 11 shows the relationship between the advance ratio J, the thrust coefficient and the pitch angle of the propeller.

The advance ratio is defined by equation 56, where V_a is the average flow speed of the water over the propeller, and can be interpreted as the distance advanced by the propeller in one revolution. It is made dimensionless by dividing it by the propeller diameter.

$$J = \frac{V_a}{nD} \tag{56}$$

One of the important effects of the propeller's interaction with the hull is thrust deduction. This is caused by a low pressure differentials on the intake side of the propeller, near the hull, which increases the resistance, and thus the necessary thrust to overcome it. This is modelled as a thrust deduction instead of as a hydrodynamic resistance. The available thrust is described as:

$$T_a = (1 - t)T \tag{57}$$

Where T_a is the available thrust, T is the generated thrust, and t is the thrust deduction number, which generally has a value between 0.05 and 0.2 [54].

Another effect produced by the hull's motion is the creation of the wake. The wake is a region of disturbed flow created by the passing of the hull in

the water. The water behind the stern of the hull acquires a forward speed producing a difference between the surge speed of the vessel and the average flow velocity over the propeller V_a, the speed of advance [5]. This difference is expressed as:

$$w = \frac{u - V_a}{u} \qquad and, \tag{58}$$

$$V_a = (1 - w)u \tag{59}$$

Where u is the surge speed of the vessel and w is the wake fraction constant which is determined empirically in a series of propulsion tests.

Although neither of these two effects has a significant impact our vessel configuration (small catamaran based vessel) they are important parameters to be considered when modelling thrust devices. Wake effects are neglected since the propeller of the vessel is set in the centreline of the platform and both hulls are set apart symmetrically at the sides.Because of this, it is assumed that there is no interaction between both hulls and the propeller. Thrust deduction effects are also neglected due to the same reasons mentioned above.

5 Model simulation

MATLAB is a software developed by the company MathWorks. Although initially the software was designed primarily for numerical computing the platform has developed a lot since the foundation of MathWorks in 1984 .This development has lead to integrated powerful tools such as Simulink.

Simulink is a block diagram environment integrated with MATLAB. Simulink is used for multidomain simulation, control design and Model-Based Design. Simulink features a set of libraries or Toolboxes that deal with specific modelling domains. The Simscape toolbox will be used in this work. Simscape is a powerful toolbox that allows the creation of complete physical systems within the Simulink environment. Simscape is divided into several subsets, allowing the user access to a wide array of tools to model systems as electric motors, hydraulic actuators, refrigeration systems, vehicle drivetrains, etc.

Simscape is divided into five subsets: Driveline, Electronics, Fluids, Multibody and Power Systems. Simscape library components represent physical elements like engines, pumps, motors and transistors. Components in the model are connected with lines that represent physical connections in the real system and transmit information or power. Depending on the type of physical domain this lines are depicted in different colours. Thanks to this approach, modelling in Simscape allows the description of the real physical system structure instead of the governing mathematical expressions. Altogether, this type of modelling quickly gives an overall sense of the complete system, simplifying the modelling process and making it more understandable.

The Multibody subset in particular will be used. Simscape Multibody is the subset of tools that provides with a multibody 3D simulation environment for mechanical systems. The Multibody toolbox supports the import of complete CAD assemblies including the metadata of masses, inertias, joints and constrains directly into the model. It also incorporates an animated 3D animation, called Mechanics Explorer, that helps debugging and visualizing the physical system.

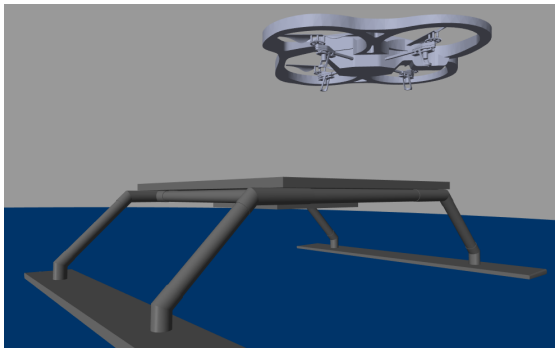


Figure 12: The AR.Drone 2.0 taking off from the Strider V1.0. Image taken from the Simulator using Mechanics Explorer.

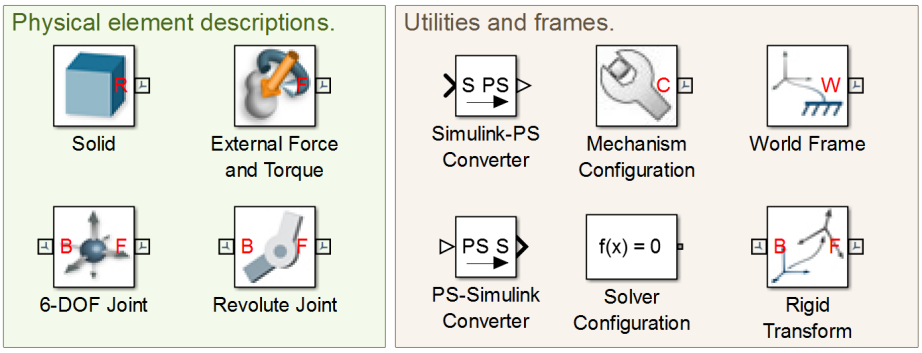


Figure 13: Some Simscape block icons.

5.1 Simscape Multibody

Simscape Multibody offers a wide array of blocks to model physical systems. The foundation of the library is the solid block, that represents a physical object. To connect and model the physical interaction and description of the systems lines and blocks such as joints, forces or frame transformations are used. Additionally configuration blocks must be added to the model network to build the simulation. Here is a short list of the most used blocks in this section. A more detailed description can be found in the available product documentation [16].

Solid This block represents a rigid solid with geometry, mass, inertia and colour properties. The solid's geometry can be defined from a list of preset shapes or can be imported along the physical properties of the solid from a STL or STEP file. Solid blocks are connected through a reference port R, that encodes the position and the orientation of the body.

Joints Joint blocks represent connections between bodies or groups of bodies with a different number of degrees of freedom. It represents the motion between a base B and a follower F frame. This frames are connected through the ports described with the same notation. Joints blocks allow the specification of different configurable parameters such as force actuation, position control or internal mechanics variables as spring stiffness of the connection or damping coefficients. The two joints shown in Figure 13 depict a 6DOF joint (full translation and rotation) and a revolute joint (rotation along the Z axis).

External Force and Torque External forces and torques are applied through the use of this block. The force or torque is applied either at the follower (port F) or the World frame upon user selection. The user can select the direction of the force and torque vector through the parameters of the block. The value of the applied force or torque can be either configured directly from the properties of the block or as an input port of the block.

Mechanism configuration This block provides the simulation parameters to a mechanism. Within this block it is possible to configure the gravitational acceleration and its direction.

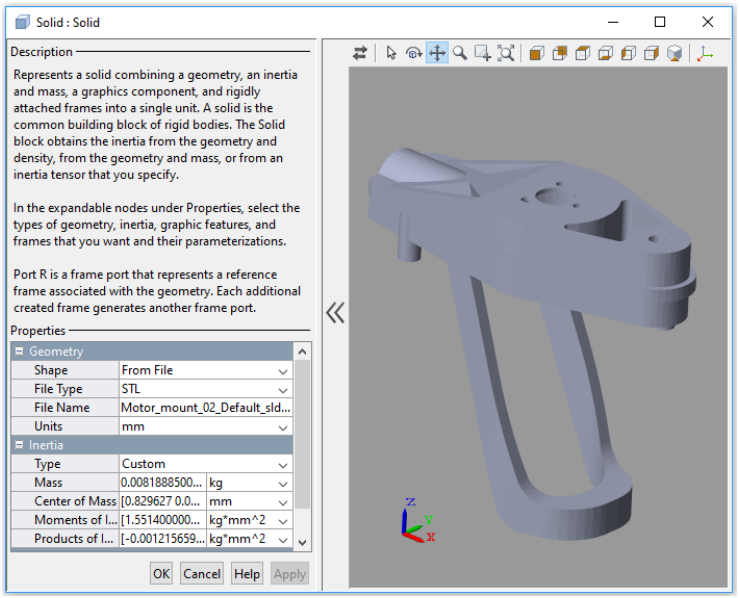


Figure 14: Solid block interface. This block represents one of the rotor mounts of a rotor assembly.

Solver Configuration Each of the physical networks defined in the model have to be connected to a solver configuration block. This block sets the solver parameters needed in order to begin the simulation. It configures parameters relevant to the simulation process such as consistency tolerance, memory budgets, sampling time, etc.

Frames The World frame block represents a unique global reference frame of the model. This frame is inertial and at absolute rest. All other frames are defined with respect to the World frame. The rigid transform block allows the rigid transformation of one frame to another. This transformation rotates and translates the frame from the follower port F with respect to the base port B frame. This translation relationship remains fixed during the simulation.

Simulink and PS converters The Simulink-PS Converter and its analogous PS-Simulink Converter convert Simulink units into physical units and vice versa. This is needed since the Simscape environment functions with physical networks and translation is needed between standard Simulink and Simscape blocks.

Within this toolset different colour codes for the connecting lines exist. Red lines represent physical units data such as Newtons, degrees, meters, etc., green lines represent physical connections between the blocks at the end of the lines while black lines represent standard Simulink data connections which meaning varies depending on the interconnected blocks.

Thanks to this approach to system modelling, the model design problem gets simplified into the calculation of the forces acting on the physical system, their

position and the interaction between the parts that conform the system. This means that the mathematical expressions of the dynamics of the systems do not need be computed, which opens the door to the simulation of very complex physical systems, where the computation of mathematical expressions for their dynamics is difficult.

For example, in the case of the quadcopter simulation, the generation of thrust is implemented through an *External Force and Torque* block that generates an upward force. The origin of this upward force is the centre of the propeller and this design is applied to the four rotors. Upon starting the simulation, the software automatically computes the motion generated by this set of forces, accounting for the physical properties of the rigid body system attached to them.

5.2 Quadrotor simulation

The quadrotor simulation is based on the CAD model shown in Figure 15. This model was submitted to the GrabCad community [7]. Some modifications were done to the CAD model, as the material properties were not configured and the mass of the quadrotor was inaccurate. Also, some geometry corrections were necessary since some elements were slightly misplaced. It is important to note that every individual piece of the assembly is implemented individually in a *Solid* block in the model. This provides a high level physical detail of the real system.

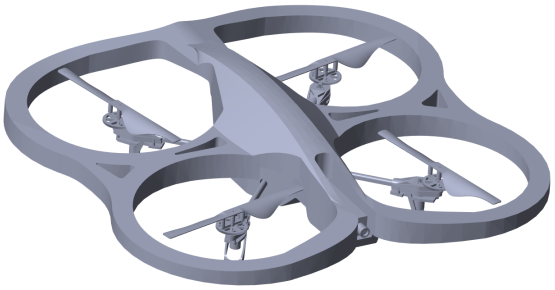


Figure 15: AR.Drone CAD model as seen in Simulink's Mechanic Explorer. CAD sourced from the GrabCad community [7].

Figure 16 shows the Simulink scheme of the quadrotor simulator. The orange subsystems and the *Path Command* block represent the control architecture of the simulation and will be covered in chapter 6. The green subsystem contains the quadrotor plant and the dynamic implementation. This subsystem receives the command for the percentage throttle of the four rotors and outputs the State Output bus signal, which contains the information about the current attitude, speed and position of the quadrotor. This signal represents the output feed of the sensory equipment of the quadrotor.

We will run through the different blocks and layers of the Quadrotor Plant

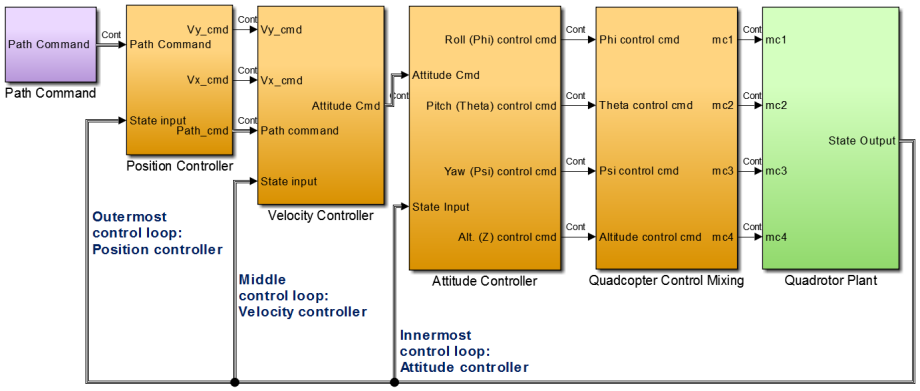


Figure 16: Quadrotor simulator scheme.

subsystem to describe the implementation of the physical system of the quadrotor into the Simulink environment.

Figure 17 shows the inside of the *Quadrotor Plant* block. Inside this block the received input is translated into a dynamic actuation on the simulated physical system. The first purple area in the scheme flow is just the conversion from percentage throttle received from the control system to the angular speed of the propeller in rad/s. This operation can be described by the expression shown in equation 60. The next purple area is the implementation of the dynamic expressions described in equations 61 and 62 , obtained from appendix A.

$$w_i = Th_{\%} \frac{RPM_{Max}}{100} \frac{2\pi}{60} \frac{rad}{s} \tag{60}$$

$$Q_i = c_D w_i^2 \tag{61}$$

$$T_i = c_T w_i^2 \tag{62}$$

Where w_i is the propeller angular speed, $Th_{\%}$ is the throttle percentage and RPM_{Max} is the maximum angular speed of the propeller in RPMs. Q_i and T_i are the generated torque drag and thrust for propeller i and c_D and c_T are the drag and thrust coefficients respectively.

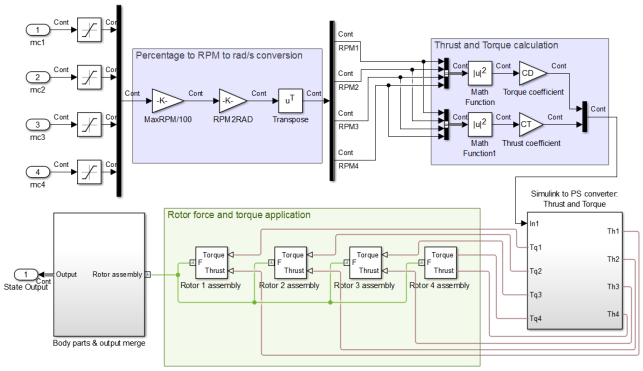


Figure 17: Quadrotor plant scheme.

Following the scheme flow, the *Simulink to PS converter* subsystem is found. It translates the Simulink signals into physical signals, compatible with Simscape blocks, by means of the *Simulink-PS converter* block (see Figure 13). This signals are fed into the rotor assembly blocks, that contain the solid parts of each of the rotor assemblies of the quadrotor. Within this block a -1 gain is introduced in two of the torques to represent the different direction of rotation of the propeller pairs.

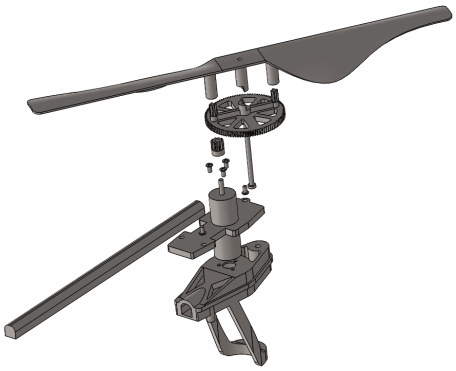


Figure 18: Exploded view of a rotor assembly.

Figure 19 shows the structure of the motor assembly block. Each of the subsystems seen in the figure contains a *Solid* block with the geometry of the individual parts that conform the rotor assembly. The same structure as the detail shown in this figure is shared by all the component parts. An exploded view of the rotor assembly is shown in Figure 18. Each of the single elements shown in the figure is implemented individually in each of the four rotor blocks, this shows the level of detail that can be achieved with this tool.

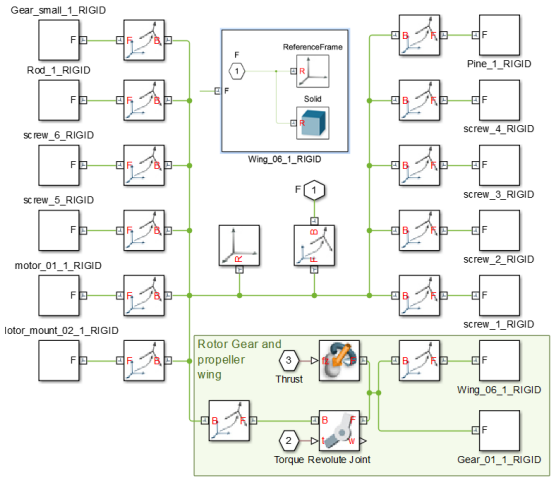


Figure 19: *Rotor assembly* bock scheme.

The *Body parts & output merge* block connects the four rotor assemblies with the rest of the quadrotor constituent parts. This subsystem also contains the simulation configuration blocks and the *World* frame block. Besides, the linear and angular damping forces, that simulate the air friction of the quadrotor body, are also implemented here. The sensing of the quadrotor attitude and position are obtained directly from the sensing of the 6DOF joint that connects the World to the quadrotor physical system. Speed readouts are obtained by deriving the quadrotor position overtime. This is done in the *Speed calculation* subsystem. All this measurements are translated from Physical signals to Simulink signals in an intermediary layer by means of the Converter blocks.

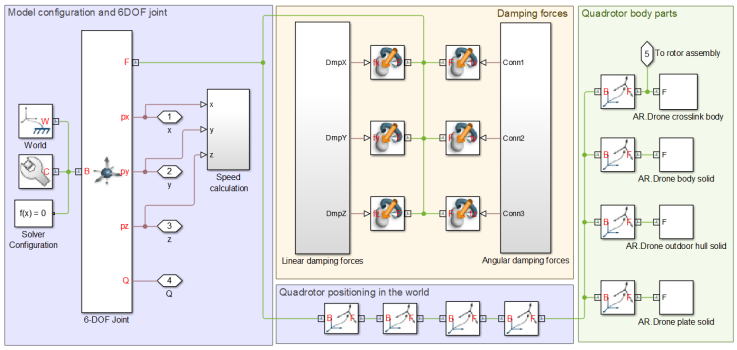


Figure 20: *Body parts* block scheme.

5.3 Strider V1.0 simulation

The Strider V1.0 simulation is based on a CAD model of the platform. It represents a simplified model of the physical system. The used CAD model of the strider is presented as a single solid instead of an assembly, as has been done with the quadcopter, lacking the same level of physical accuracy the quadcopter model provides. However, since this model's purpose is the design of a suitable

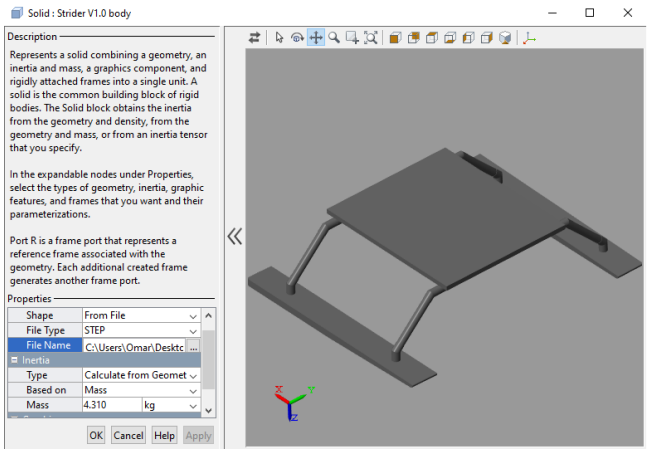


Figure 21: Strider V1.0 *Solid* block showing the CAD model used for the simulation.

control system in a 3DOF manoeuvring approach it can be assumed that this will have little effect on the simulation results. This assumption can be made based on the simplifications made in chapter 4.2.2.

As in the previous section, we will run through the model's architecture to describe the implementation of the vessel dynamics into the Simulink environment.

Figure 22 shows the simulator architecture. The system presents a similar structure to the quadrotor simulation although it has a more simple control architecture. The orange and purple subsystems are covered in section 6 as they contain the controller implementation. We will then explore the Plant subsystem to describe how the dynamic implementation of the model was made.

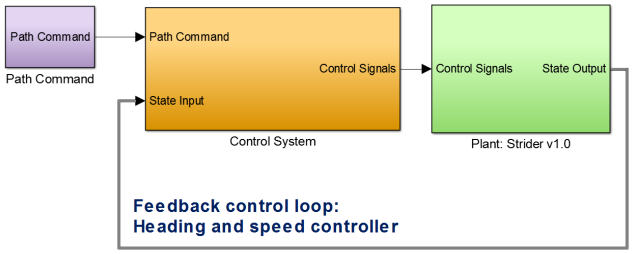


Figure 22: Strider simulator architecture.

As seen in Figure 23 the Strider Plant is less complex than the quadrotor plant. Considering the vessel as a single solid body considerably helps with this simplification, along with having only two actuators instead of the four present in the quadcopter. The purple area seen in the figure contains the sensing of the vessel attitude and position along with a small subsystem that uses blocks to implement the expression shown in equation 63 in order to translate the vessel

speed from the inertial frame to the body fixed frame.

$$\begin{bmatrix} u & v \end{bmatrix}^T = \boldsymbol{R} \begin{bmatrix} \dot{x} & \dot{y} \end{bmatrix} \tag{63}$$

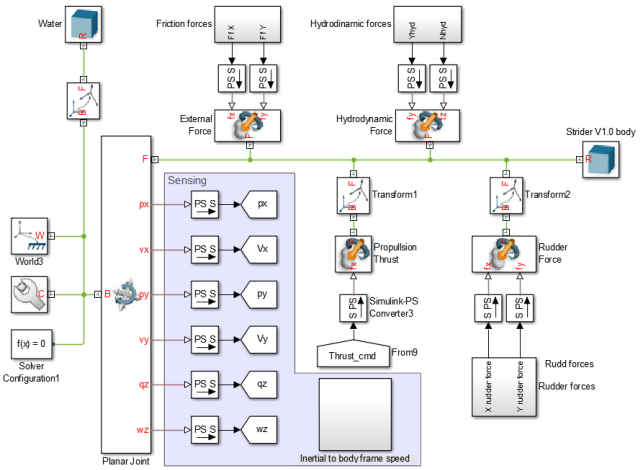


Figure 23: Strider plant architecture.

Recalling from section 4.2 the forces at play in the vessel had three main sources, namely hydrodynamic forces, external forces and control forces. In the plant model the four components of these forces can be seen (control forces have been divided in rudder forces and propeller thrust). The four *External Force and Torque* blocks seen in the figure are feed by four subsystems. This subsystems are just the block diagram implementation of the expressions shown below:

• **Hydrodynamic forces** are modelled after the linear approach made by Davidson and Schiff in equation 45. The added mass terms have been neglected based on the assumptions made in the dynamics' study. The rudder component included in this expression has been also neglected since a more precise rudder model will be implemented separately.

$$Y = Y_v v + Y_r r \tag{64}$$

$$N = N_v v + N_r r \tag{65}$$

Figure 24 shows the *Hydrodynamic forces* block. It is important to note that since the hydrodynamic derivatives where calculated for each hull, the contribution of the two hulls needs to be added, thus the multiplier added at to the output of the block. Due to the small size and draft of the hulls and the separation between them, we can simplify the problem and assume that there exists no interaction between the hulls, doubling the actuation of the generated forces.

• **Rudder forces** are modelled after equation 51 and are applied at the coordinates of the centre of pressure $\boldsymbol{CP^B}$ of the rudder. The subsystem contains the block representation of the below equations:

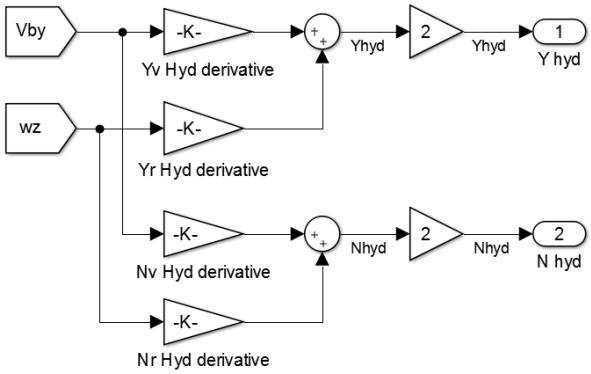


Figure 24: *Hydrodynamic forces* block.

$$\begin{aligned} X_{rudder} &= -F_R(u, V_{av}, v, r, \sigma)sin(\sigma) \\ Y_{rudder} &= F_R(u, V_{av}, v, r, \sigma)cos(\sigma) \\ Z_{rudder} &= 0 \end{aligned} \tag{66}$$

where the expression for the rudder force F_R is:

$$F_R = \begin{cases} \frac{1}{2}\rho C_L A_r V_{av}^2 sin(\frac{\pi}{2}\frac{\sigma_{attack}}{\sigma_{stall}}) & if|\sigma_{attack}| < \sigma_{stall} \\ \frac{1}{2}\rho C_L A_r V_{av}^2 sign(\sigma_{attack}) & if|\sigma_{attack}| \geqslant \sigma_{stall} \end{cases} \tag{67}$$

• **Friction forces** are modelled using the common drag equation for motion in a fluid:

$$F_{ext} = \frac{1}{2} C_f u^2 \tag{68}$$

Where C_f is the friction coefficient and u is the speed of the body. The value of C_f is obtained from the work of [3].

• **Propeller thrust** has been modelled as a direct command since neither the wake factor nor the thrust deduction produced by the interaction of the propeller with the hull. This assumption is explained section 4.2.3 where propulsion forces are covered.

5.4 Parametrization of the simulation

For both the quadrotor and the vessel simulation, solid geometry, mass and inertia properties have been directly imported from the existing CAD models. Table 6 provides with the data concerning the rest of variables not parametrized in the CAD models such as lift and drag coefficients and the value of the hydrodynamic derivatives.

Platform	Variable	Value
AR.Drone	Propeller thrust coefficient	$9.14\dot{1}0^{-6}$
	Propeller drag coefficient	$2.38\dot{1}0^{-9}$
	Air drag coefficient	0.1
	Max. propeller rpm	5000 rpm
Strider V1.0	Y_v	-3.89950
	Y_r	1.77427
	N_v	-1.77427
	N_r	-0.80729
	$x_{cp} - x_{cg}$	0.347 m
	Air drag coefficient	0.2
	Rudder lift coefficient	1.2
	Rudder area	0.0064 m^2
	Water density	998 $Kg\ m^{-3}$

Table 6: Simulation Parameters.

Some of the parameters of the simulation have been obtained from external sources. Detailed sourcing and comments on them is given below.

The data for the maximum RPM of the rotors along with the values for the thrust C_T and drag C_D coefficients have been obtained from [36] and [37]. Both works present with empirical results from model identification tests of the AR.Drone 2.0.

The values of the hydrodynamic derivatives have been obtained from the expressions in table 12 from Appendix B:

$$Y_v = -\frac{\pi}{2}\rho U T^2 \tag{69}$$

$$Y_r = \frac{\pi}{4}\rho U T^2 L \tag{70}$$

$$N_v = -\frac{\pi}{4}\rho U T^2 L \tag{71}$$

$$N_r = -\frac{\pi}{8}\rho U T^2 L^2 \tag{72}$$

Where $L = 91\ cm$ is the hull length, $T = 5\ cm$ is the draft depth, $U = 1\ m\ s^{-1}$ is the forward speed and $\rho = 994\ Kg\ m^{-3}$. This measurements have been obtained directly from the CAD model used in the simulation.

Rudder lift coefficient is estimated based on the works of [68], where a general guideline based on experimental results gives good estimations on lift coefficient in spade rudders.

Air drag coefficients for the vessel model have been obtained from the work made during the development of the Strider V1.0 prototype [3].

6 Control system

This chapter will cover the control techniques implemented and the system control architecture. First, PID feedback techniques are illustrated and then the control architecture of the quadcopter and the vessel are explained.

6.1 PID control

PID control techniques are the most used industrial linear regulators. This is due to their simple structure and easy implementation, good applicability to a wide arrange of control problems and their tunability of "blackbox" systems where the plant is not identified.

PID controller stands for proportional-integral-derivative controller and is a control method based on the combination of proportional, integral and derivative contributions of the error value. The error value $e(t)$ is the difference between the reference setpoint and the actual measured value of the controlled variable. The controller's aim is to minimize the error overtime by means of a control signal which expression is:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de(t)}{dt} \tag{73}$$

Where K_p, K_i and K_d are the proportional, integral and derivative gain coefficients respectively. Each contribution to the control signal has different effects on the response of the system and is controlled by means of the PID gains mentioned above [69].

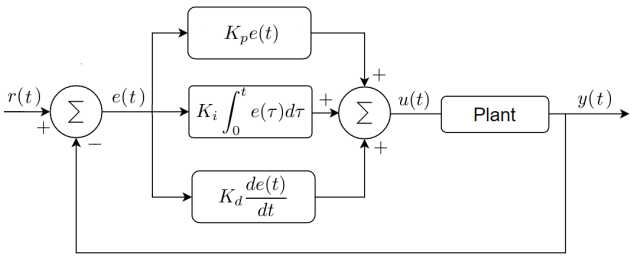
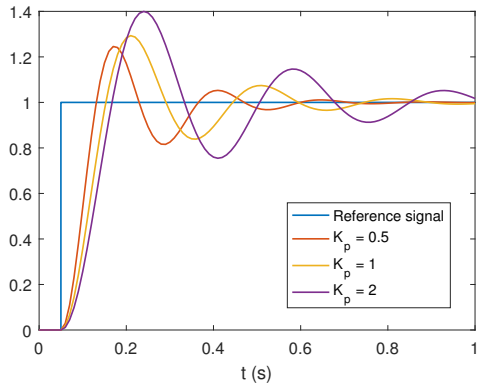
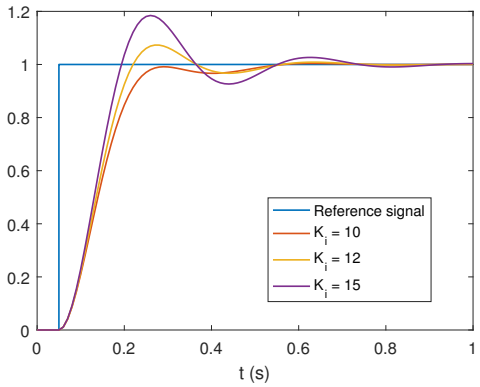


Figure 25: PID block diagram. $r(\tau)$ is the desired setpoint and $y(\tau)$ is the measured value of the controlled variable.

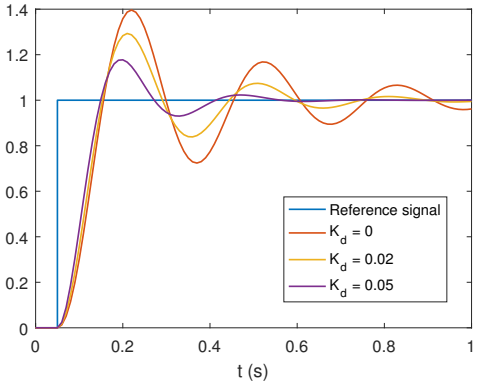
- **Proportional term:** proportional gain accounts for the present values of the error. The proportional term multiplies the present value of the error by K_p. High values of K_p produce rapid changes in the value of the error, however it increases the overshoot of the system response and can rend the system unstable. On the other hand, low proportional gains lead to less sensitive and slow controllers.
- **Integral term:** integral gain accounts for the past values of the error. This term is the sum of the instantaneous errors over time and multiplies it by the K_i coefficient. Although integral action increases the overshoot



(a) Kp gain variation response for a step input. Both K_i and K_d remain constant.



(b) Ki gain variation response for a step input. Both K_p and K_d remain constant.



(c) Kd gain variation response for a step input. Both K_p and K_i remain constant.

Figure 26: Effects of proportional, integral and derivative gains in system response

and the settling time, this contribution eliminates the steady state error. However, excessive integral actuation on the system can lead to integral windup caused by large setpoint variations causing an unstable response.

- **Derivative term:** derivative gain accounts for the rate of change of the error over time. Derivative actions predicts the behaviour of the system and stabilizes the system along with a reduction of settling time. One of the problems associated with derivative action is the necessity of adding low pass filters for the input of this term as high frequency noise has great impact in the value of the derivatives.

Figure 26 shows how the variation of the three PID gains affects the transient and steady state response of the system.

The designed control architecture for the quadrotor control consists in a series of cascade PID feedback loops. This allows for a finer and easier control, apart from the flexibility derived from the ability to tune specific parts of the

loop.

Cascade loops benefit from dividing the control problem into several parts, adding complexity to the overall system but simplifying the individual nested loops. Cascade architecture generally results in a quicker and smoother control and it is one of the most recursive methods to improve disturbance rejection performance, although it comes with drawbacks. It usually involves additional sensory equipment, however that is not relevant in our working platforms since IMUs already measure all the necessary parameters required to achieve control of the system without the need of additional equipment. Another inherent problem of cascade systems is the extra tuning necessary to implement the control laws, although this tuning is usually straight-forward and easy to do. Figure 27 illustrates the cascade architecture used in this work for the control of the quadcopter.

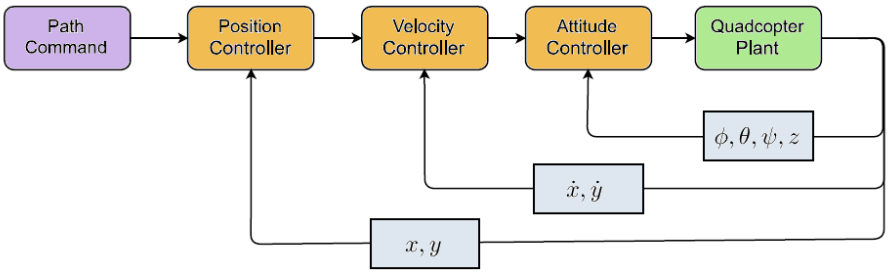


Figure 27: Cascade loop architecture used for the control of the quadcopter.

6.2 PID tuning and system requirements

Many approaches for PID tuning have been developed over the years. Formulas to derive PID controller parameters are result in a wide variety of tuning methods. Among the most popular methods we find the Ziegler-Nichols method (with its two variations), the Cohen-Coon method, IAE-setpoint, ITAE-setpoint, internal model control, etc. Many works such as [70] or [71] evaluate some of the most used PID tuning techniques and analyse their relative performance and robustness between them.

However, the increased computing capabilities brought by the advances in data processing lead to the creation of PID tuning software and developments in automated PID loop tuning. This techniques usually yield better results than traditional PID tuning techniques, specially in systems with long response times where other tuning methods may take up very long times to simply get a stable set of parameters. There exist many algorithms and approaches to software tuning that allow to tune the control loops to different criteria. This feature, along with the capability of the software to automatically optimize tuning solutions are the reasons why most modern industrial applications use this type of tuning methodology.

In this work the PID Tuner algorithm implemented in one of the toolboxes

available in Simulink will be used. Simulink Control Design is an integrated toolbox that allows the design and analysis of control systems modelled in Simulink. It features a PID tuning algorithm that is controlled via a graphical interface. The algorithm objectives are focused on closed-loop stability, adequate performance based on loop frequency bandwidth and adequate robustness, based on gain and phase margin allowance.

The control interface of the tuner is accessible from the PID blocks of the system and can be seen in Figure 28. This interface presents with several options that enable fine tuning of the system response. The algorithm lets you set a design focus, either reference tracking or input disturbance rejection, and allows tuning of the system in both time and frequency domain by means of a set of sliders. The sliders control the response time and robustness of the system by modifying loop bandwidth for the first and gain and phase margin for the second.

Since automatic tuning requires linear plants, the tuner tries to analyse all blocks between the output and input of the PID controller block and generate a linearised plant around the initial conditions specified in the Simulink Model. Although in simple models the tuner is able to produce a linearised plant of the system straight away, in the case of more complex systems, the tuner is not able to produce a plant from the block analysis. In this cases the PID Tuner includes several options to obtain a suitable linear plant:

Import linear plant. It is possible to import linear plants from the MATLAB workspace that could have been obtained beforehand. i.e from experimental data or previous plant analysis.

Re-linearise around another operating point. The tuner will relaunch the block analysis around a selected time in the simulation. This is valuable when steady state conditions do not hold at the initial moment of the simulation and steady state tuning is necessary.

Identify a new plant. A new plant is identified from the simulated output of the system. When this option is selected, the user is prompted to chose an input signal for the system (step, impulse or custom input signal). Then the complete model is simulated in an open loop configuration and the output of the system is registered. A plant identification is run on this registered data and a linear transfer function for the plant is presented.

This last method was the one used in this work since model block linearisation was not possible due to the complexity of the modelled systems. This linear approximation presents some minor accuracy problems in cases like the quadcopter model, where the strong non-linear dynamics of the system affect the quality of some linearised models. In the particular case of the plant identification of the quadcopter, the obtained PID parameters provided by the tuning algorithm resulted in system responses slightly off the predicted ones by the tuning software. This problem was corrected by minor manual corrections on the provided PID gains based on some engineering judgement and iteration.

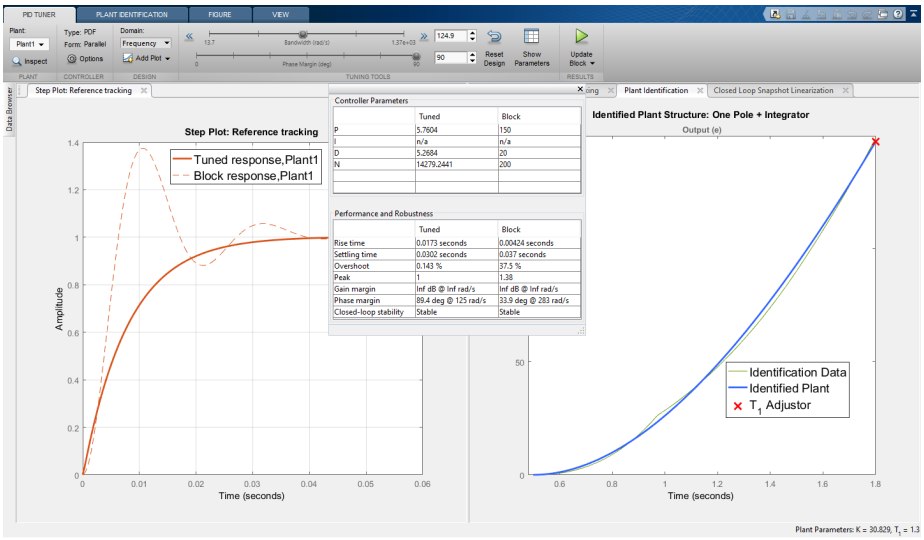


Figure 28: PID tuner graphic interface. The Figure shows the plant identification and response tuning of the quadcopter's roll.

System requirements

The system's requirements for the control design of both the quadrotor and the vessel where aimed at a robust system performance. The robustness of the systems is of crucial importance because of the high impact of external disturbances and measurement noise that the two platforms will be subject to when operating. A robust control system will also help the performance of the real system when encountering plant parameter variations and unmodelled plant dynamics. The impact of unmodelled plant dynamics is particularly important when considering all the linear approximations and simplifying hypothesis made during the dynamic modelling of both plants. The linearisation performed by the automated plant identification used for PID tuning also introduces uncertainties in the model.

For the quadrotor, the control architecture is aimed at fast attitude responses with wide gain and phase margins for good input disturbance and noise rejection. The reasoning behind this strategy is based on the chosen control architecture. In cascade architectures the transient response of the inner control loop directly affects the stability of the outer loop. Since attitude responses are directly bound to velocity and hovering control, fast attitude controllers are paramount for the stability of the quadcopter. Due to this system architecture, saturation of the control signals for each of the controlled parameters is necessary, since high proportional gains are needed to guarantee that the adequate time response is achieved without compromising the robustness of the system.

The trade off of this strategy is the longer transient response of the altitude control. Due to the limited thrust resources of the quadcopter, fast altitude commands to the rotors would quickly saturate the four actuators, leaving no room to attitude control commands and rendering the quadrotor completely

unstable due to the lack of available thrust to achieve the desired attitude.

On the other hand, thanks to the simplicity of the control architecture of the vessel, the control requirements for this system only have to aim at a robust architecture without any other consideration. Wide gain and phase margins for input and noise disturbances controllers are implemented. The limitations of this control architecture are set by the maximum thrust of the propeller and the maximum rudder angle. Rudder turning speed will be neglected due to its high value (see Table 5) and the propeller dynamics will also be neglected, assuming the instant generation of thrust, due to the small scale of this contribution to the dynamics of the vessel.

6.3 Trajectory tracking: Way-point guidance

Planned routes of any unmanned vehicle can be represented in terms of way-points. Way-points are defined in Cartesian coordinates (x_k, y_k, z_k) for $k = 1, 2, \ldots, n.$ and represent an ordered database of points in the working space. This way-point databases can be expressed as [64]:

$$wpt.pos = (x_0, y_0, z_0), (x_1, y_1, z_1), \ldots, (x_n, y_n, z_n) \tag{74}$$

Additional parameters can be added or subtracted to the way-points information. We will take up onto the case of surface vessel guidance to exemplify this. In surface vessel trajectory tracking only the (x_k, y_k) are considered, but other parameters like speed or heading can be included in the way-point variables. For example, a way-point could be set to have a predetermined position (x_i, y_i) and set a surge speed U_i and heading angle ψ_i required when passing through it.

One of the simplest and common methods of implementing path control based on way-point trajectory planing is the use of Line Of Sight(LOS) guidance. LOS guidance is based on the calculation of a straight trajectory from the current position of the vehicle and the following way-point. The desired heading angle ψ_d is obtained using the following expression:

$$\psi_d(t) = tan^{-1}\left(\frac{y_d(k) - y(t)}{x_d - x(t)}\right) \tag{75}$$

Where $\psi_d(t)$ is the desired course angle, $y_d(k)$ and $x_d(k)$ are the next way-point coordinates and $y(t)$ and $x(t)$ the current vehicle position. An important remark concerning equation 75 is the necessity of a proper quadrant selection mechanism. Since trigonometric functions only provide solutions for the I and IV quadrant a sign test has to be performed to determine the desired heading angle ψ_d quadrant.

Once the vehicle has reached the way-point the next way-point is selected. For this purpose, the concept of circle of acceptance is adopted. When the vehicle resides within the borders of a circle of radius $\rho_0[m]$ the next way-point in the database is selected. This condition is translated into the following inequality derived from the Pythagoras theorem.

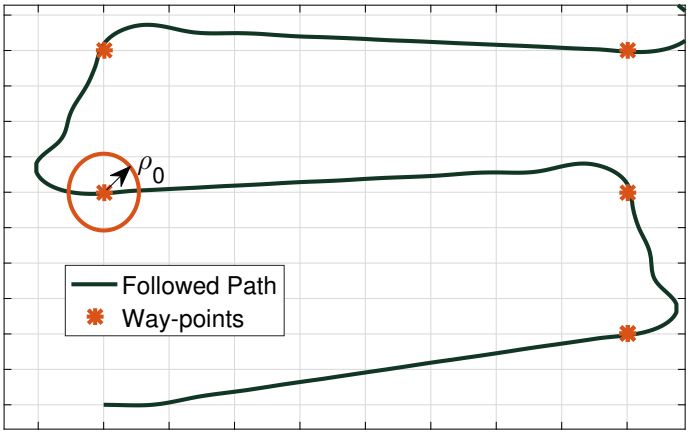


Figure 29: Way-point trajectory guidance based on LOS.

$$[x_d(k) - x(t)]^2 + [y_d(k) - y(t)]^2 \leq \rho_0 \tag{76}$$

When this condition is met the next way-point $(x_d(k+1), y_d(k+1))$ is selected. A general guideline for the selection of ρ_0 is to set it equal to one or two ship lengths, i.e. $\rho_0 = L$ *or* $2L$ [54].

6.4 Quadrotor control system

As seen section 4.1, control of the quadrotor is achieved by the individual control of the speed of each of the four rotors. By differentially controlling the speed of different sets of rotors the four basic movements of the quadrotor can be obtained, namely throttle T, roll ϕ, pitch θ and yaw ψ.

Figure 27 shows the architecture used in for the control of the quadcopter and Figure 16 its block implementation in Simulink. Three nested feedback loops are used, dividing the control problem into three different sections. The attitude controller corresponds to the innermost nested loop and feeds the Control Mixing block where the control signals are translated into percentage throttle for each rotor. The middle control loop corresponds to the Velocity controller block. The velocity controller feeds the inner loop the desired attitude to achieve the setpoint speed. The outer loop is connected to the Position Controller and is fed by the path command. It sets the velocity needed to achieve the desired position.

Figure 30 shows how the translation from throttle, roll, pitch, yaw to the individual rotors percentage thrust is made in the *Quadcopter control mixing* block. Equation 77 describes the expression for each one of the rotors. Note that the rotor notation is the same used in Figure 5.

$$\begin{aligned} mc_1 &= Alt_{cmd} - \psi_{cmd} - \theta_{cmd} \\ mc_2 &= Alt_{cmd} - \psi_{cmd} + \phi_{cmd} \\ mc_3 &= Alt_{cmd} + \psi_{cmd} + \theta_{cmd} \\ mc_4 &= Alt_{cmd} + \psi_{cmd} - \phi_{cmd} \end{aligned} \tag{77}$$

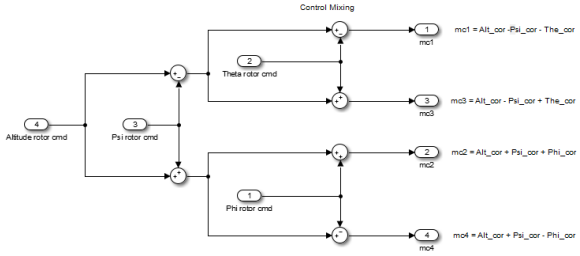


Figure 30: *Quadcopter control mixing* block

Where mc_i is the percentage rotor command to rotor i, Alt_{cmd}, ψ_{cmd} and θ_{cmd} are the outputs of the innermost control loop.

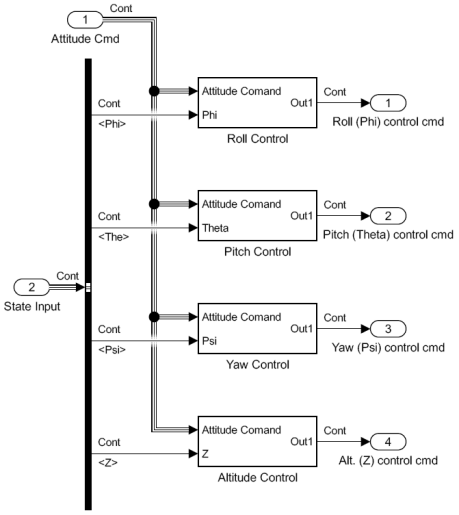


Figure 31: Attitude controller

Attitude controller The attitude controller is the innermost nested loop and receives the current values of ϕ, θ, ψ and z from the State Input vector and the attitude command ($\begin{bmatrix}\phi_{cmd} & \theta_{cmd} & \psi_{cmd} & z_{cmd}\end{bmatrix}$) from the velocity controller. Four PID controllers have been implemented for the roll, pitch, yaw and altitude control. Each of the smaller control blocks features the same structure as Figure 32. In this blocks, the pertinent variable is extracted from the Attitude command bus, the error between the actual and desired value is calculated and the PID is implemented. After this, the control signals are sent to the control mixing to translate this output to individual throttle commands to the rotors.

Roll, pitch and yaw control feature only proportional-derivative (PD) gains while the altitude controller adds the integral action to the PD gains to eliminate steady state error in a full PID structure

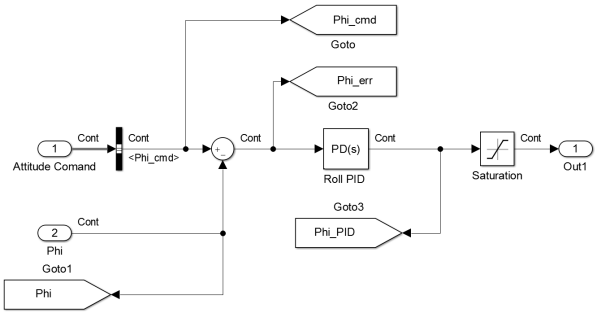


Figure 32: Roll PID control structure

The saturation implemented after the control signal in Figure 32 is added to avoid rotor saturation. Due to the characteristics of the system, changes in pitch and roll drastically affect the stability of the quadrotor. For this reason, fast responses are needed to achieve precise control of the quadrotor, this translates into high proportional and derivative gains that, while performing well when the error is small, make the system unstable with big changes in the setpoint value. To neglect this phenomena the output control signal of the PIDs is saturated, achieving the desired fast system response.

Table7 gathers all the information concerning PID gains and saturation values of the Attitude controller.

	K_p	K_i	K_d	Saturation [Min, Max]
Roll	175.3	-	19.5	[-20, 20]
Pitch	199.72	-	20.04	[-20, 20]
Yaw	15.17	-	6.98	[-10, 10]
Altitude	170.24	28.86	101.52	[45, 90]

Table 7: Attitude Controller PID parameters.

Velocity controller The Velocity Controller has a similar structure to the Attitude Controller. The Velocity Controller acts as a pass-through for the ψ and z commands and implements a PD controller with the same structure as Figure 32 for the u and v velocities. The input of this PID loop are the desired velocities Vy_{cmd} and Vx_{cmd} (both expressed in terms of the body frame) and its outputs are the needed ϕ and θ angles to achieve this velocity.

Table 8 gathers all the information concerning PID gains and saturation values of the Velocity Controller.

Position controller The position controller involves a PI control and it also includes the necessary calculations to obtain the position error. However, in this case the error calculation is not a as trivial as in the previously mentioned

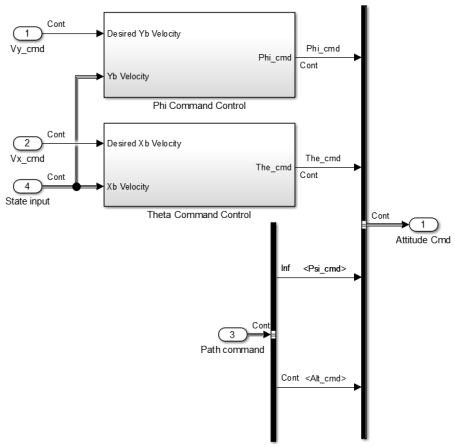


Figure 33: Velocity Controller

	K_p	K_i	K_d	Saturation [Min, Max]
Y Speed	19.66	-	0.259	[-15, 15]
X Speed	26.94	-	-	[-15, 15]

Table 8: Velocity Controller PID parameters.

controllers. The complications resides in the fact that changes in the yaw angle produce a misalignment between the body and inertial frames. This implies that the velocities in the inertial and body frames are not coincident anymore. This problem can be solved by translating the received x^E and y^E positions from the State Input bus (which are the inertial frame x and y coordinates) to the body frame through a simplified two dimensional expression of equation 8 in terms of the yaw angle ψ:

$$\mathbf{R} = \begin{bmatrix} cos\ \psi & -sin\ \psi \\ sin\ \psi & cos\ \psi \end{bmatrix} \tag{78}$$

then,

$$\mathbf{\Gamma^E} = \mathbf{R}^{-1} \begin{bmatrix} x^B \\ y^B \end{bmatrix} \tag{79}$$

Equation 79 is implemented in the *Position Error* block as seen in Figure 35. Thanks to this transformation control of the Velocity and Attitude Controllers is straightforward as u and v velocities are directly controlled by the pitch and roll angles respectively.

Table 9 gathers all the information concerning PID gains and saturation values of the Velocity Controller.

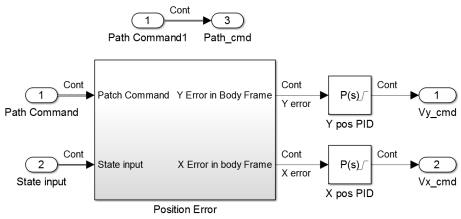


Figure 34: Position Controller

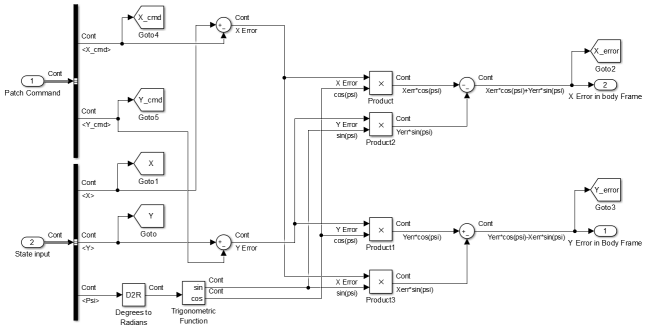


Figure 35: *Position Error* block. This diagram represents equation 79 implemented graphically.

	K_p	K_i	K_d	Saturation [Min, Max]
Y Position	2	1.1	-	[-2.5, 2.5]
X Position	2	1.1	-	[-2.5, 2.5]

Table 9: Position Controller PID parameters.

Path Command As its name suggests, the *Path Command* block outputs the desired position of the quadrotor with respect to the inertial frame and the yaw and altitude command values. It feeds the position controller the desired x and y position with respect to the inertial frame. This commands are issued from the workspace, but can also be set directly with a Simulink block. Figure 36 shows the inside of this block.

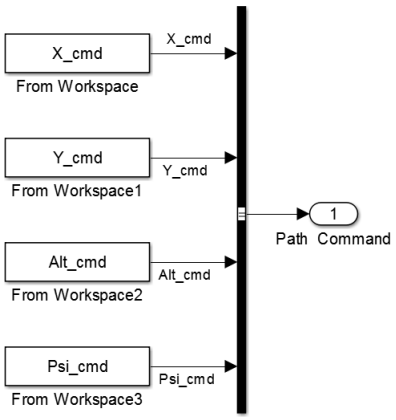


Figure 36: *Path command* block.

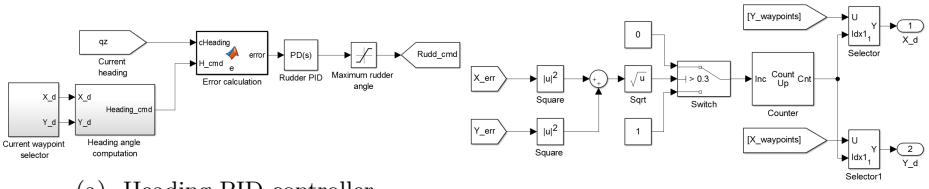
6.5 Strider V1.0 control system

The control architecture of the Strider V1.0 is easier to implement than the quadrotor system. The system has two inputs: the rudder angle δ_r and the generated thrust T, compared to the four of the quadrotor. Besides this, only two degrees of freedom of the vessel will be controlled. Two separate PID controllers will command the surge speed u and the heading ψ of the vessel. A way-point guidance control is used for trajectory planning.

Heading controller Heading control is achieved by means of rudder angle δ_R variation. A PD controller is used in order to achieve this. Figure 37a shows the controls scheme used. The saturation block used limits the maximum rudder angle δ_{max}.

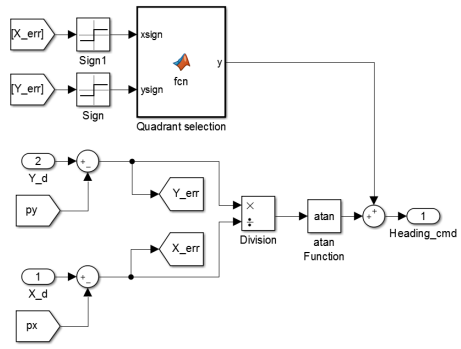
In Figure 37a two additional blocks can be seen. The *Heading computation* block graphically implements equation 75 along with a MATLAB function to perform the quadrant selection. The *Current way-point selector* block is equivalent to equation 76 and performs the shifting of the way-point database when the listed condition in said equation is met.

Error calculation is not as straightforward as in the previous PID implementation because of the periodicity of the heading angle measurement (i.e $30 = 30 + k360, \forall k \in \mathbb{Z}$). In order to avoid the problems derived with this periodicity and the angle notation hustle, for example set the error between 10 and 350 equal to 20 instead of 340, the Matlab function *Error calculation* is introduced for optimization of the controller. The code listing of this little algorithm is shown in Listing 1.



(a) Heading PID controller .

(b) *Heading angle computation* block.



(c) *Current way-point selector* block.

```
function y = fcn(xsign,ysign)
if xsign >= 0 && ysign >= 0
    y = 0;
elseif xsign >= 0 && ysign < 0
    y= 2*pi;
else
    y = pi;
end;
```

(d) *Quadrant selection* code listing.

Figure 37: Heading control architecture

Listing 1: *Error calculation* block code listing

```
function [error]= e(cHeading,H_cmd)

% periodicity calculation
k = floor(abs(cHeading)/(2*pi));

% correcting values over 360 and below -360
if cHeading >= 0
        heading = cHeading-k*2*pi;
else
        heading = cHeading+(k+1)*2*pi;
end

% compute the shorter angle to the desired heading.
if abs(heading-H_cmd) > pi && heading-H_cmd < 0
        error = 2*pi+(heading-H_cmd);
elseif heading-H_cmd >= pi && heading-H_cmd >= 0
        error = (heading-H_cmd)-2*pi;
else
        error= heading-H_cmd;
end
```

Speed controller Speed control is achieved by means of a PI controller. The integral gain is needed to correct the steady state error produced in the system. Figure 38 shows the block diagram of the speed controller. The saturation ap-

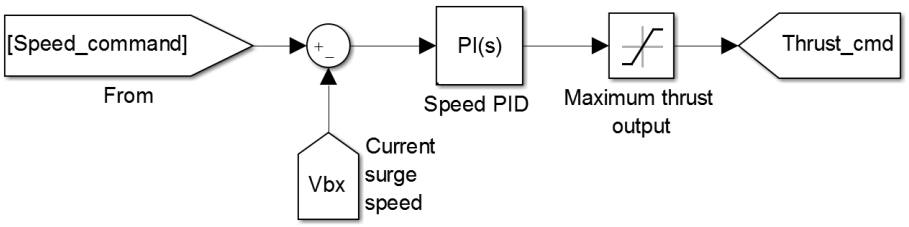


Figure 38: Speed PID controller.

plied at the output of the control signal represents the maximum thrust output of the propeller.

Table 10 gathers all the information concerning PID gains and saturation values of the speed and heading controllers.

	K_p	K_i	K_d	Saturation [Min, Max]
Speed controller	82.65	-	-	[-3, 3]
Heading controller	3.826	-	0.4601	$[-\frac{7\pi}{36}, \frac{7\pi}{36}]$

Table 10: Position Controller PID parameters.

The saturations implemented in these blocks correspond to the maximum thrust that the propeller can provide and the maximum turning angle of the rudder.

Path Command Similarly to its quadrotor homonym the *Path Command* block outputs the desired position, in this case the way-point vectors, and the commanded forward speed. The commands can be issued from the workspace, but again, can also be set directly with a Simulink block. Figure 39 shows the inside of this block.

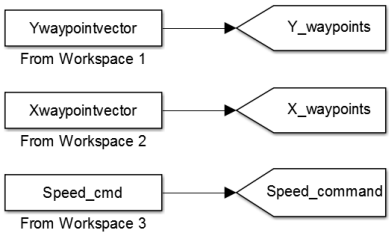
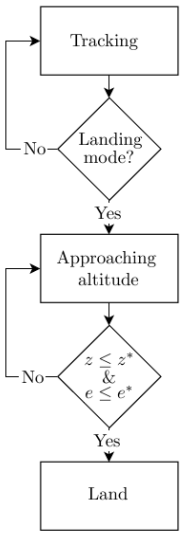


Figure 39: *Path command* block.

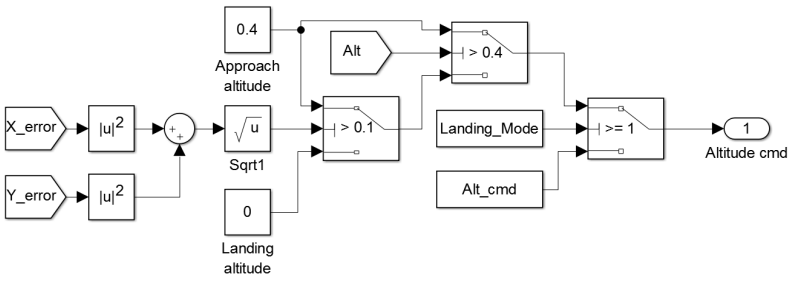
6.6 Landing on a mobile platform

Autonomous landing on a mobile platform proves to be a difficult task because of the involved complexity of precise position estimation. The usual approach to this problem is the use of Visual Servoing for position and tracking control as works such as [72] and [73] show. Although a specific control design based on Visual Servoing is out of the scope of this work, the performance of VTOL tasks with the designed controller can be evaluated.

The design of the autonomous landing controller is based on the simple flow chart shown in Figure 40a. During the approaching phase, the quadrotor is set to track the vessel at an altitude z^* over the landing platform. If the position error e is less than the selected threshold e^*, the quadrotor is commanded to land. Figure 40b illustrates the Simulink block diagram used to implement this controller that feeds directly the altitude command.



(a) Controller flowchart



(b) Controller block diagram

Figure 40: Flowchart and block implementation of the autonomous landing controller.

7 Simulation results

This chapter will present the results obtained from the simulation of several tests. Results from VTOL on moving platforms, vehicle tracking, way-point guidance and hovering will be presented. The chapter is structured in several sections showing the results of the different performed tests.

For a more realistic simulation and assessment of the control architecture robustness noise was added to the measurement signals. Limited band Gaussian white noise was used for the modelling of this disturbance. This noise has a normal distribution with zero mean μ_n and finite variance σ_n^2, where σ_n is the standard deviation. The noise frequency selected is 200 Hz since this is the transmission frequency of the drone and one of the commonly used ROS data rates. The modelling parameters for the noise are shown in table 11. The noise parameters for the AR.Drone's sensory equipment have been obtained from [74] and [36].

Parameter	$\boldsymbol{\sigma_n}$
Roll	0.0716°
Pitch	0.0859°
Yaw	0.0517°
Altitude	0.0017 m
Speed	0.01 $m\ s^{-1}$
Position	0.01 $m\ s^{-1}$

Table 11: Standard deviation values for the modelling of noise for the different sensory equipment of the AR.Drone.

Way-point guidance

Way-point guidance was implemented for position control of the Strider V1.0. Several tests were performed with different path setups. A constant speed of 1 m/s was commanded to the vessel since this is the reference speed used for the dynamic modelling of the forces. The below figures show the results obtained in this tests. White noise with a standard deviation $\sigma_n = 0.015$ was introduced to simulate the measuring error of the velocity measurement and evaluate the controller. For the Heading angle error a $\sigma_n = 0.017$ was used. The values of the noise for this two parameters were estimated since there was no available data.

The constant speed and the maximum rudder angle limit the maximum turning angle of the vessel. This is the main reason of the overshoots between closeby way-points seen in the tests. Another important parameter of way-point guidance control is the value of the acceptance distance ρ_0. Figure 41 shows the different responses of the system to various ρ_0. As it can be seen from the figure, higher radius of acceptance yield smoother paths since the task is less demanding and the selection of the next way-point is made earlier. However,

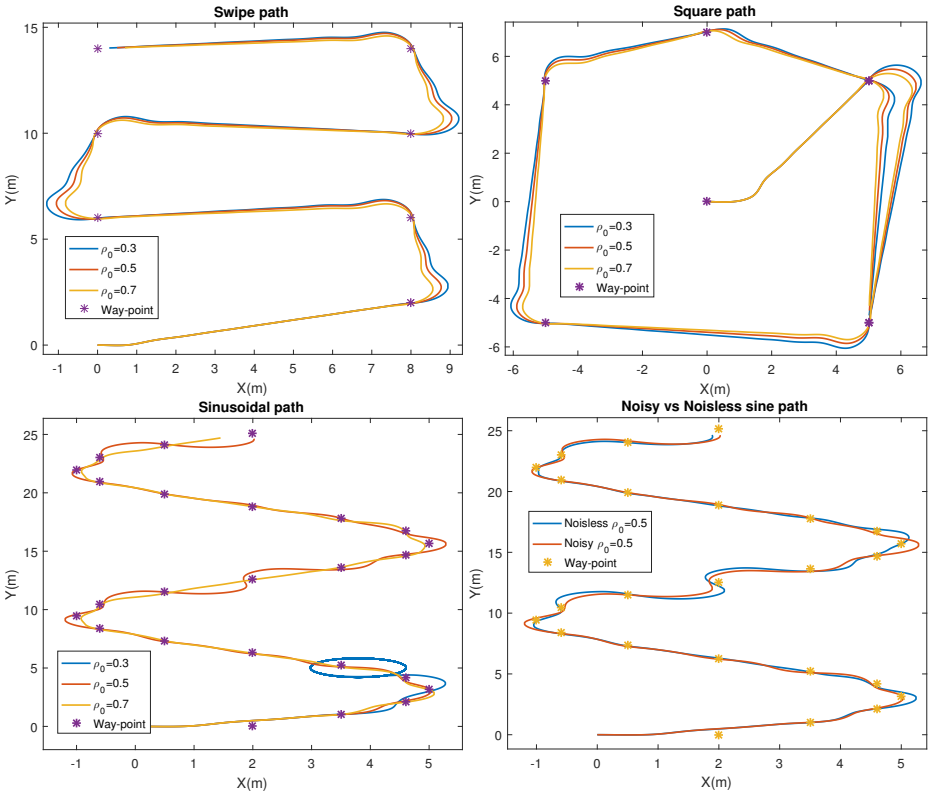


Figure 41: Path following performance of the Strider V1.0

this results in less precision in close series of way-points and is not suitable for tasks where high precision is needed. On the other hand, short ρ_0 yield closer passes by the way-points, but incur in higher overshoots as can be seen in the Swipe ad Square path figures.

The sinusoidal path plot presents one of the problems derived from short radius of acceptance. As can be seen in the bottom left image of Figure 41, due to the high overshoot in the first curve of the path, the vessel gets stuck in a circling pattern. The limited turning capabilities of the vessel, due to stall rudder angles and constant speed, and the short ρ_0 render the system incapable of reaching the next way-point.

The last plot in Figure 41 shows the performance difference between a noiseless test and a test where white noise was added. Although the system is affected by the measurement disturbances, the robustness of the implemented controller allows the system to perform the task similarly to the unaffected vessel. Figure 42 shows the performance comparison of the two implemented PID loops in a swipe path test when subjected to noisy measurements. The results show a small lag in the control response of the system but the stability of the control loops is not affected. The top plot of the figure also illustrates the correct functioning of the heading command optimization made in the control architecture.

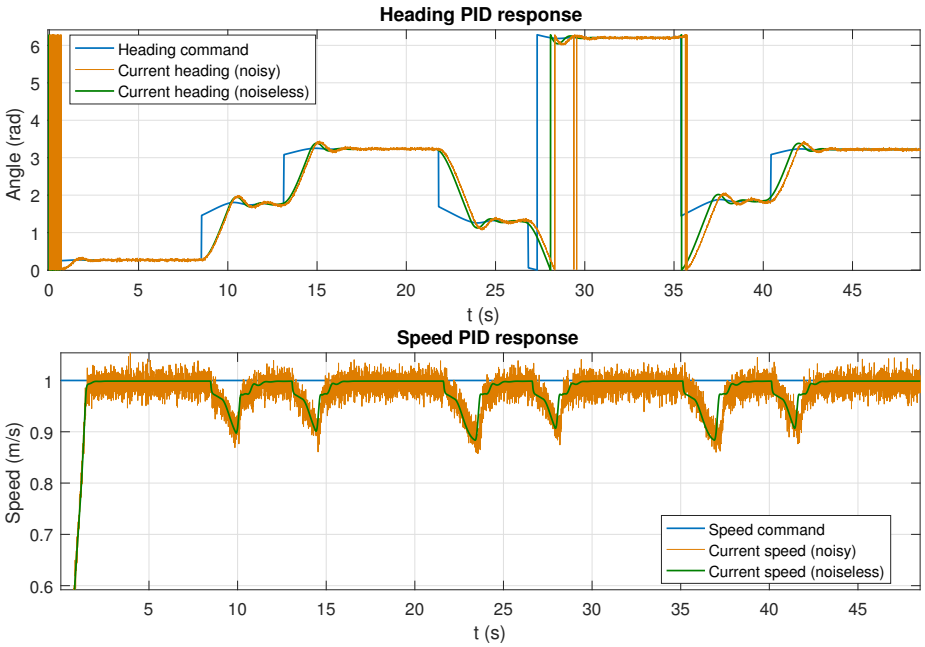


Figure 42: PID performance in a swipe path test ($\rho_0 = 0.5$).

Hovering

Quadrotor hovering abilities are one of the most important characteristic of the platform. The robustness and stability of both roll and pitch are very important since an angle different than zero produces motion in the horizontal plane and this difficults position keeping. Simulations to evaluate the hovering performance and robustness of the system under measurement noise were performed. This measurement noise is modelled as a Gaussian white noise using the values shown in Table 11.

The quadrotor is tasked to fly to a determined location and hover. Figure 43 shows the position keeping performance of the system when subject to noise while performing this task. As it can be seen from the results of the tests, the disturbance input does not affect the stability of the system and the position error stays within 5 *cm*.

Roll and pitch angles during the test are displayed in Figure 45. The values present small errors once the stable position is obtained, with less than one degree of error of the commanded attitude and close to zero values.

Altitude and yaw values are shown in Figure 44. The big spike at the beginning of the yaw angle plot is due to the momentary saturation produced by the big control outputs of the altitude, pitch and yaw. However, once the altitude error gets reduced and the actuators are "freed" quick control of the yaw angle is obtained, with a less than 1 degree error once the yaw is stabilized. The altitude plot of the hover test shows a damped response, designed in this

manner instead of going for a quicker under damped response of the system to avoid a worse saturation effect as the one shown in the yaw plot and mentioned in chapter 6.4.

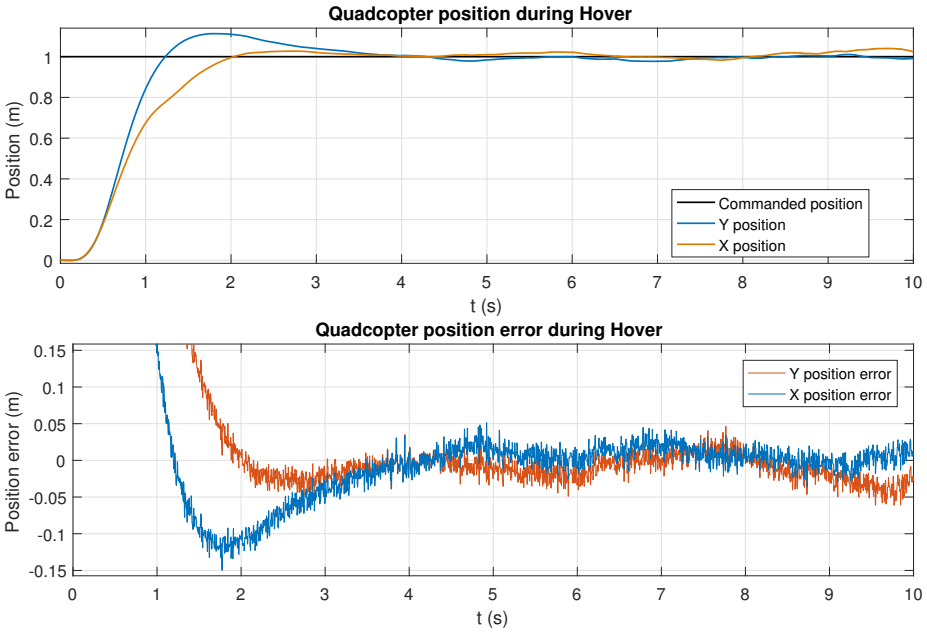


Figure 43: Position and position error in the Hover test.

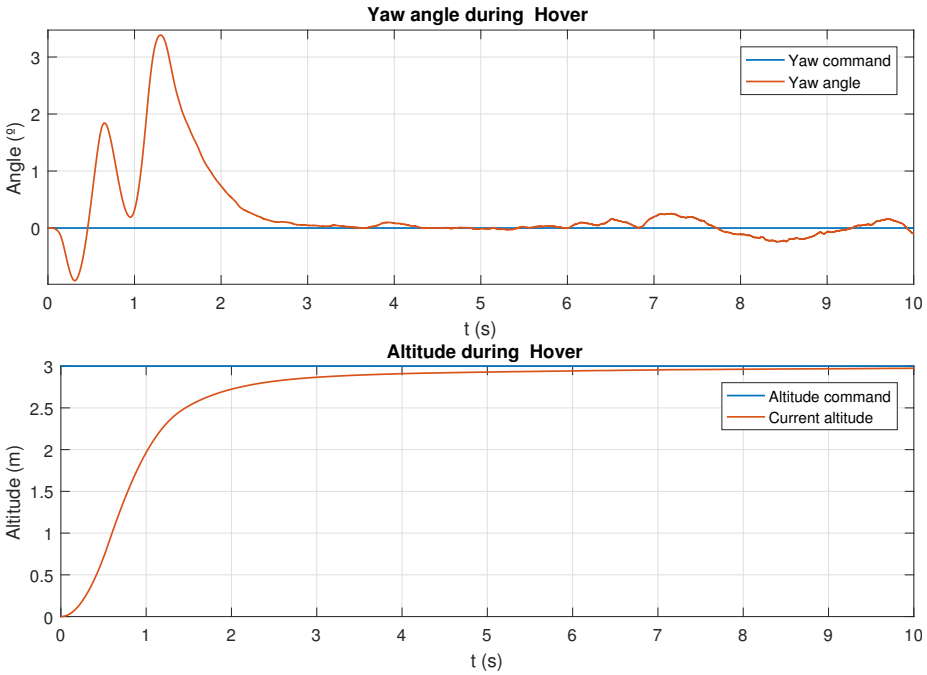


Figure 44: Yaw and altitude responses in the Hover test.

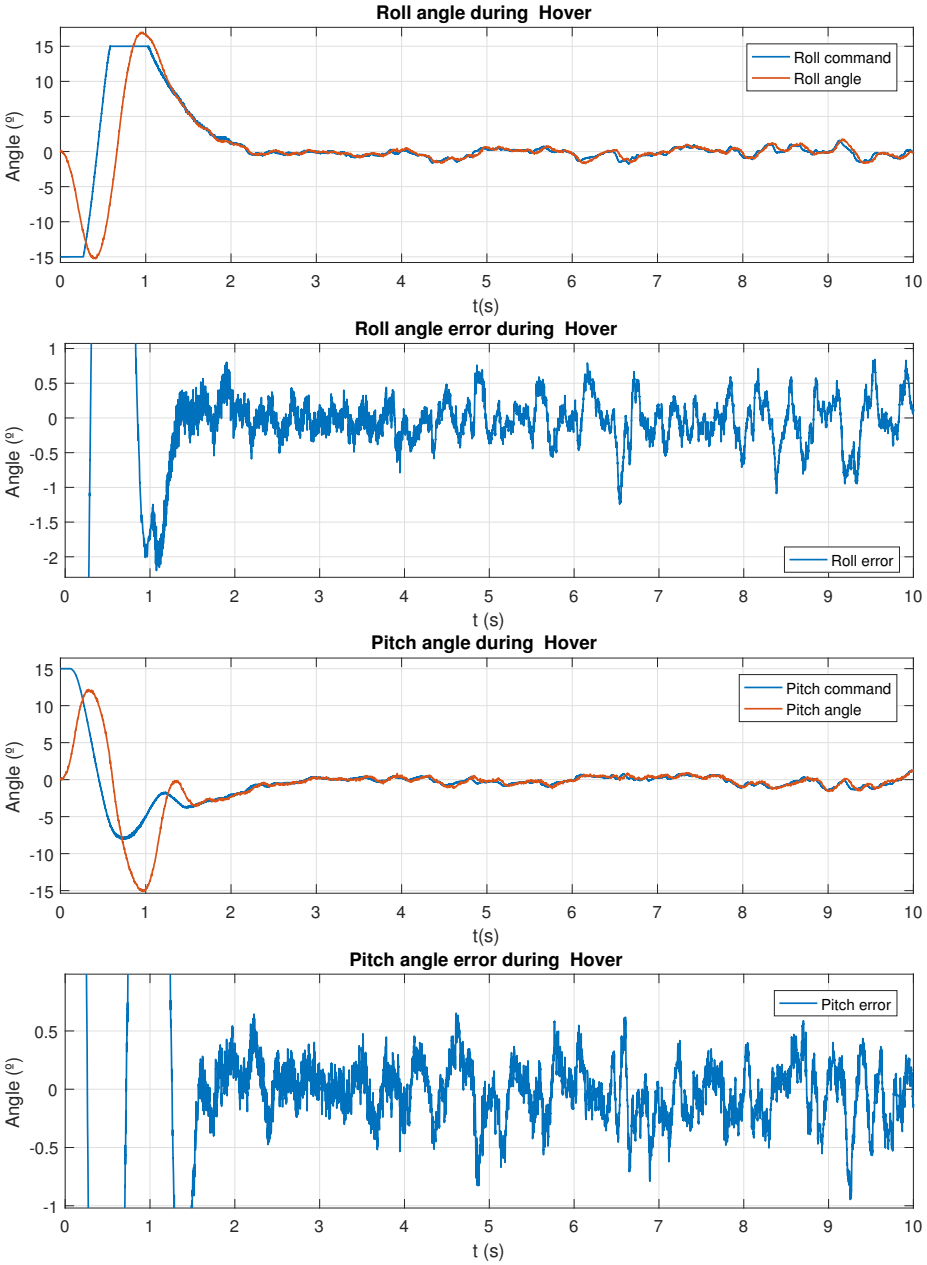


Figure 45: Roll and pitch angles and errors in the Hover test.

Tracking

In this test the quadrotor is set to track the position of the Strider V.1. while the vessel performs path following tasks.

The performance of the controller can be evaluated considering the position error between the quadrotor and the Strider V1.0 shown in Figure 46. The sim-

ulation results show that the position controller is sensible to sharp turns and responds slowly to direction changes. Nonetheless, the performance in straight paths is really good and the position error stays within a 5 *cm* distance emulating the results seen in the hover test.

One of the main reason of this high overshoot behaviour in turns is the lack of friction damping of the quadcopter. This, paired with the unattainability of faster controllers due to actuator saturation imply that the tracking performance worsens the faster and sharper the turns are.

Because of this, the controller would be heavily benefited by the application of predicting control algorithms that take into account the future values of the path in order to perform this tasks more efficiently.

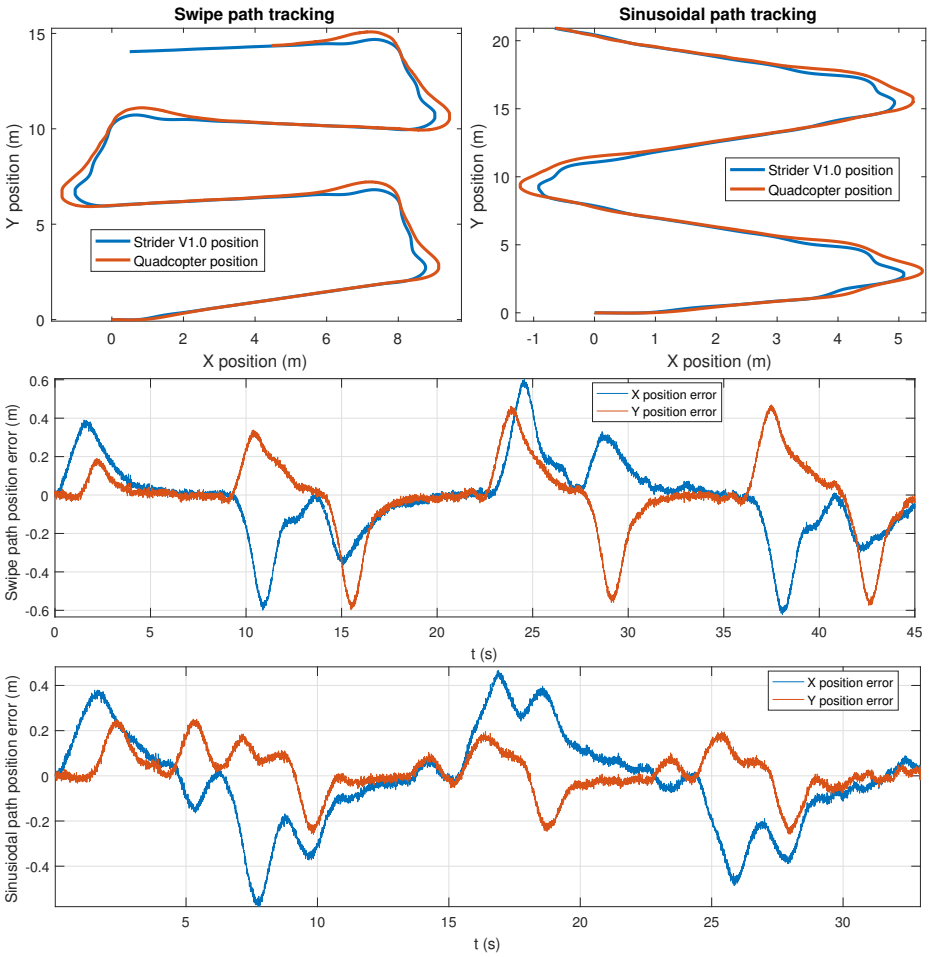
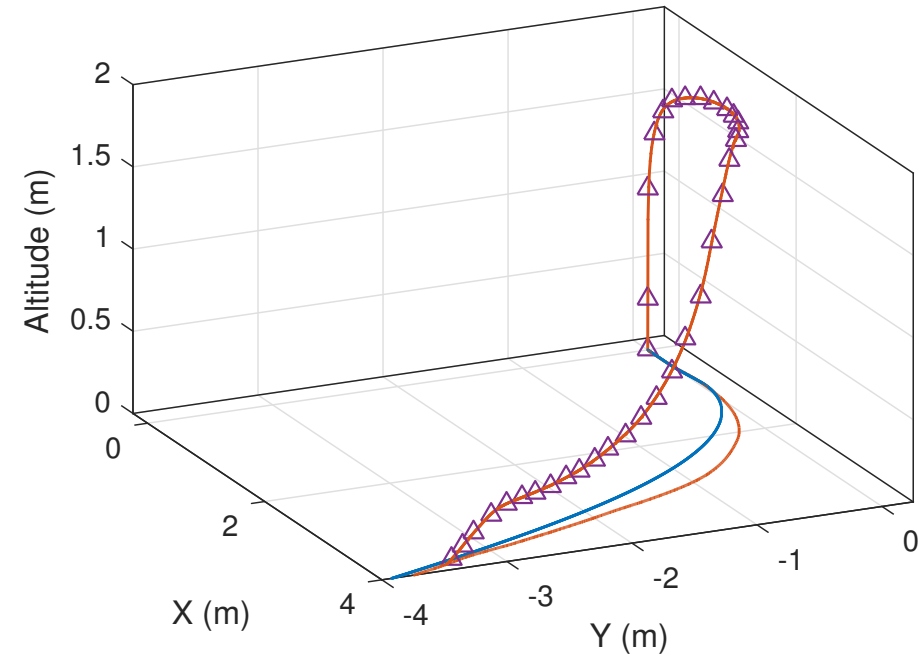


Figure 46: Tracking performance and position error of different paths.

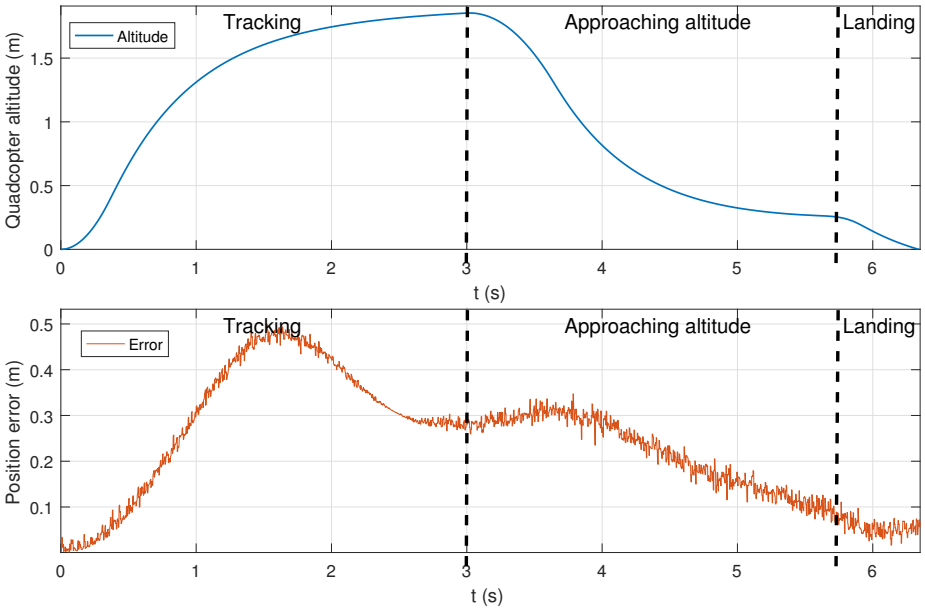
Autonomous VTOL

In this test the developed controller in section 6.6 is used to perform an autonomous VTOL flight. The controller parameters used for the test were $e^* = 0.1\ m$ and $h^* = 0.3\ m$.

The results of the performance of the VTOL tests are shown in Figure 47. It can be seen that the designed algorithm functions properly and that the quadcopter is able to safely land in the platform within the error constrains that were set. In Figure 47a the blue line represents the position of the Strider V1.0, the marked orange line represents the position of the quadcopter while the simple orange line represents the horizontal projection of the quadcopter position. The plots in Figure 47b show the altitude and the value of the error e during the test. In the plots, the three stages of the controller can be easily identified and are separated in different areas.



(a) 3-D plot of the autonomous VTOL flight.



(b) Altitude and position error of the quadcopter.

Figure 47: Results from the autonomous VTOL test.

8 Conclusions

This thesis sets working ground for the use of the new Simmechanics environment and is a starting point for dynamic modelling and control design based on Model Based Design for research applications.

Following the set objectives of this thesis a dynamic model of the quadcopter AR.Drone 2.0 and the Strider V1.0 unmanned surface vehicle have been obtained. With this dynamic models a suitable simulation environment was developed using Simscape Multibody toolset within Simulink. This simulation environment was then used for the design of a control architecture based on PID feedback loops to achieve control of both platforms and perform different tasks.

Dynamic modelling of the UAV and USV were presented in chapter 4 and Appendixes A and B deepened in dynamic modelling related topics. The simulation environment used in this work is presented in chapter 5 and the modelling architecture illustrated. Control system architecture is covered in chapter 6 along with some of the basis of PID control design, trajectory tracking and autonomous VTOL. Finally chapter 7 presents the results obtained from the simulations carried out to asses the performance of the designed control.

The obtained simulation results validate the designed control architecture. Way-point guidance control tests for the Strider V1.0 show great performance even when subject to measurement noise and the effect of different acceptance radius is noted. Tracking performance of the quadcopter yields positive results with low position errors and quick response to direction changes. The hovering capabilities of the designed controller were also tested, showing great attitude stabilization, presenting only small angle errors and great position keeping abilities. This tests proved the robustness of the quadcopter against input disturbances. VTOL tests on the moving landing platform were also performed producing satisfactory results.

Although the simulation gave satisfactory results, ongoing work and several features to be added in the future for the improvement of the reliability of the simulator are listed below.

The physical model of the Strider V1.0 could be improved to a more detailed model with the empirical calculation of the hydrodynamic derivatives. This would open the door to more complex but accurate non-linear dynamic models and the addition of environmental contributions such as wave induced forces. Besides this, a more complete path following algorithm could be implemented to increase the complexity of path following tasks and avoid some of the drawbacks derived from line of sight guidance.

Another improvement to the model could be the inclusion of the actuator dynamics for both the quadcopter rotors and the thruster and rudder mechanisms of the vessel. For an even more precise model, the sensory equipment and data communication could also be introduced in the model, expanding the reach of the simulation and providing a more complete working environment. More specifically, a camera model could be added to the simulation to emulate

visual servoing control and directly evaluate the efficacy of position and velocity tracking.

Model validation of the simulation results and the model parameters based on empirical data would improve the reliability of the designed control for the system. Besides this, it would be interesting to compare the performance of PID control to other control approaches, easily found in the literature, and make a comparison analysis to analyse the correlation between performance and complexity of the different control architectures.

Appendices

A Aerodynamic forces

This appendix deals with the calculations of thrust and torque generated by the propeller rotation due to aerodynamic forces. It also covers other aerodynamic effects such as blade flapping and induced drag.

A.1 Aerodynamic forces: Thrust

The rotors of the quadcopter have three basic functions. The generation of vertical force (thrust), the generation of horizontal force to provide motion in this plane and the means to control the attitude of the quadrotor.

Momentum Theory can be used to model the aerodynamic forces produced by the propeller's rotation. Momentum Theory is based on the following assumptions:

- The air is considered a perfect gas and incompressible
- The vertical speed of the air stream through the disk is continuous.
- The propeller is considered an infinitely thin disk that offers no resistance to the passing air.
- The flux stream of air crossing the disk is considered without interaction with the external air.
- Purely one dimensional analysis

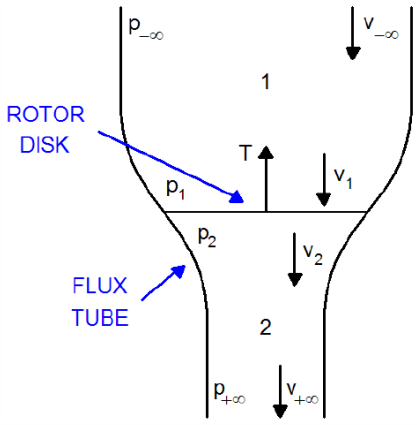


Figure 48: Momentum theory diagram. From [8].

The lift generated by the rotor can be defined as the variation of air momentum through the disk. This thrust will be proportional to the pressure

difference through the propeller. Depending on the approach taken to describe it, the thrust can be expressed as [8]:

$$T = A(p_1 - p_2) \tag{80}$$

$$T = \dot{m}_A(v_{-\infty} - v_{+\infty}) = \rho_A A v_1(v_{-\infty} - v_{+\infty}) \tag{81}$$

Where T is the generated thrust, A is the area of the disk, $\dot{m}_A$ is the variation of air mass through the disk and ρ_A is the air density. p_1 and p_2 are the air pressures directly over and below the disk while $p_{+\infty}$ and $p_{-\infty}$ are the air pressures asymptotically over and under the disk respectively. Flow speeds $v_1, v_1, v_{+\infty}$ and $v_{-\infty}$ follow the same convention as the pressure notation.

Then, using Bernoulli equation between sections $-\infty$ and 1 and sections 2 and $+\infty$ the following expressions are obtained:

$$p_{-\infty} + \frac{1}{2}\rho_A v_{-\infty}^2 = p_1 + \frac{1}{2}\rho_A v_1^2 \tag{82}$$

$$p_2 + \frac{1}{2}\rho_A v_2^2 = p_\infty + \frac{1}{2}\rho_A v_\infty^2 \tag{83}$$

Considering the inflow speed of the air v_I to be:

$$v_I = v_1 - v_{-\infty} = \frac{v_{+\infty} - v_{-\infty}}{2} \tag{84}$$

Where, after rearranging equations 82 and 83, and considering that $p_a = p_{-\infty} = p_\infty$, the speed v_1 is:

$$v_1 = \frac{v_{+\infty} + v_{-\infty}}{2} \tag{85}$$

Taking into account equation 81 (based on momentum variation) and all the above manipulations the generated thrust can be expressed as:

$$T = 2\rho_A A v_1 v_I \tag{86}$$

If hover or near hover conditions are assumed, a safe assumption to be made when high speed motion or acrobatic manoeuvres are not on expected, $v_{-\infty} = 0$ can be set, meaning that $v_1 = v_I$. Besides this, relating the inflow speed with the angular propeller speed yields the following expression:

$$v_I = \lambda w_i r_i \tag{87}$$

Where λ is the inflow ratio and relates the inflow speed to the rotor tip speed, w_i is the rotor angular speed and r_i is the radius of the propeller. Using this expression within equation 86 yields the below definition of thrust:

$$T = 2\rho_A A \lambda w_i r_i \tag{88}$$

The coefficient c_T can be introduced so the final description for the propeller generated thrust is a simpler expression:

$$T = c_T w_i^2 \tag{89}$$

Where c_T can be easily determined by static thrust test and w_i is the propeller's angular speed.

A.2 Aerodynamic forces: Drag

In a broad sense, Drag can be defined as resistance to the motion of a body in a fluid. The induced torque on the rotor of the quadcopter is produced due to the drag created by the blade rotation.

Drag production in propeller blades can be divided into two main contributions [75].

- **Profile drag**, that is geometry dependent and is caused by the loss of energy caused by the turbulent wake produced by the separation of flow from the surface of the body. It also includes the drag contribution produced by the skin friction between the contact layer of the fluid and the body.
- **Induced drag**, due to the pressure differential that exists between the two faces a propeller wing. When this two pressure areas join at the end of the wing tips and the trailing edge of the wing, a vortex is produced, creating a downward airstream in the vicinity of the blade. This causes the generated lift to be inclined backwards, which horizontal component contributes to the drag force.

The analytical expression for Drag torques can be derived from the application of blade element theory combined with some momentum theory concepts. However, following the work in [76], with the basis set by the momentum theory development made previously and dimensional analysis techniques drag torque can be expressed as:

$$Q_d = \frac{1}{2} c_d \rho w^2 \tag{90}$$

As done previously with thrust modelling, a lumped parameter model for the rotor torque produced by the drag can be presented:

$$Q_d = c_D w_i^2 \tag{91}$$

Where the coefficient c_D can be easily determined by static drag tests and w_i is the propeller's angular speed.

A.3 Other aerodynamic forces: Induced Drag and Blade Flapping

Some of the most significant aerodynamic effects associated with rotor crafts are induced drag imbalance and blade flapping. Despite not being usually included in simulation models because of their small contribution (although they are important in full sized rotor design) they are relevant effects because they produce forces in the horizontal plane of the quadrotor.

When a rotor is moving in the horizontal direction a thrust imbalance is produced. The origin of this thrust imbalance is the difference in flow speed between the advancing and retreating blade. Figure 49 illustrates this effect in a helicopter rotor.

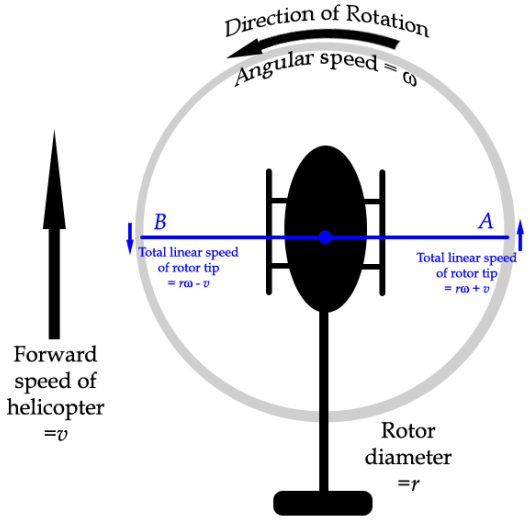


Figure 49: Lift imbalance mechanism illustrated in a helicopter rotor blade.

The blade moving in the direction of the aircraft is considered the advancing blade while the retreating blade is the one moving in the opposite direction of the aircraft. The advancing blade generates more lift as the total linear speed experienced by the wing is higher than the retreating blade.

This dissymmetry of lift not only induces gyroscopic effect but it also imbalances the induced drag generated by the propeller. While in hover conditions, the induced drag is only responsible of the rotor drag torque since its horizontal contribution gets cancelled out. However, due to the aforementioned lift dissymmetry the net induced drag force opposes the direction of the apparent wind, generally the vehicle's speed direction, developing a horizontal force.

Blade flapping is an effect also produced by the horizontal translation of a rotor through the air and the dissymmetry of lift. This thrust imbalance generates a moment in the rotor. However, considering the propeller as a spinning disk, it is easy to see that the propeller acts as gyroscope and a 90^{o} phase lag on this moment is observed. This tilts the rotor plane backwards and creates a horizontal contribution from the generated lift of the propeller. An equilibrium angle is attained because the advancing blade decreases its angle of attack as it raises, and thus its lift, countering the additional lift that would have been generated by the increased tip velocity. The reversed case occurs for the retreating blade, where the increased angle of attack is countered by the decreased tip velocity. Figure 50 illustrates this effect.

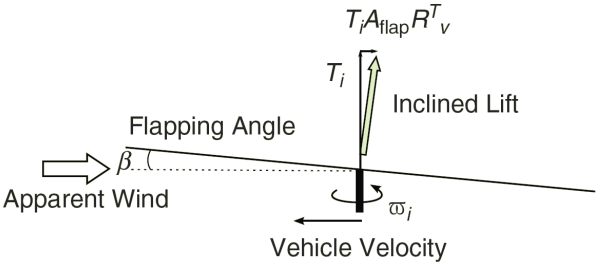


Figure 50: Blade flapping effect. From [9].

B Vessel modelling and hydrodynamic forces

This appendix covers hydrodynamic related topics. It will cover the two different traditional approaches to ship modelling, namely, seakeeping and manoeuvring theories. It will also cover the source of hydrodynamic forces acting on a moving hull and finally the modelling of said hydrodynamic forces by means of the hydrodynamic derivatives, along with some common approaches to their calculation.

B.1 Model classification

A definition for Model classification depending on its complexity or application can be found in Fossen's Marine Hydrodynamic handbook [64]. This models can be sorted into three types:

Simulation Model Simulation models focus on accuracy and generally use a full 6DOF approach. These models include marine craft dynamics and environmental forces (wind, waves...) along with modelling of the propulsion and measurement systems. Simulation models should be able to simulate on trigger events like failures, accidents or erroneous signals while matching the real system time responses.

Control Design Model These models feature reduced order or simplified dynamic simulations. Their main objective is to provide a reasonable accurate framework in the selected degrees of freedom in order to design motion control systems. Generally they are used to compute the necessary control parameters of to test specific control systems to be applied to the model.

Observer Design Model Observer models can be considered a simplified version of the simulation model. They focus on the modelling of sensors, navigational systems and disturbances. Usually observer models approach the modelling of wave, wind and ocean current forces as coloured noise disturbances.

Depending on the approach made in the analysis of the hydrodynamic forces and moments a further distinction is made in Fossen's work between two main theories. Both theories are based in different assumptions, making them more suitable for specific types of applications. These theories can be used in any of the different types of model mentioned above[64].

Manoeuvring Theory Manoeuvring Theory involves the study of the ship's movement at a constant or slowly varying positive speed. A three degrees of freedom approach is commonly considered, where only surge, sway and yaw are analysed. Restricted, calm water and still waves are assumed. The theory's basis resides in the assumption that the coefficients related to the description of hydrodynamic forces are frequency dependent. Zero wave excitation (calm water) implies that these coefficients can be represented as constant parameters. Manoeuvring theory presents simple methods based in linear derivation of these coefficients while more complex models will derive these coefficients through methods as Taylor series expansions or cross-flow drag.

The zero wave excitation assumption can only be carried to surge, sway and yaw motions because of their close to zero natural frequency [54]. The other three remaining degrees of freedom, namely, heave, roll and pitch, present higher natural frequency values, thus making them unreliable, since they violate the basic assumption of the theory and would present non realistic simulation results.

These models are usually used for autopilot design, course control and dynamic positioning. Sometimes they include roll motion to describe coupled lateral motions more accurately.

Seakeeping Theroy While Manoeuvring theory has its basis on the calm water assumption, Seakeeping theory studies the motion of the vessel when there is wave excitation, while the craft's speed and heading is assumed to be constant. Although both theories study the motion, stability and control of surface crafts seakeeping's approach gives a better framework for capability and operability analysis of the studied vessels. Seakeeping also introduces fluid-memory effects as an additional dissipative force actuating on the hull as an result of the change in pressure and momentum of the surrounding fluid made by the hull generated waves [77].

In seakeeping, hydrodynamic coefficients and wave forces are computed as a function of the frequency of the excited waves. This values are also dependent on hull geometry and mass distribution

These models are usually used for vessel stability studies and the aforementioned capability and operability analysis

B.2 Hydrodynamic forces

Hydrodynamic forces acting on the hull of a vessel are assumed to be linearly superposed [78], thus allowing the differentiation between the different contribution the hydrodynamic force is made of. Depending on the originating effects it can be divided as [5]:

- **Motion in an ideal fluid with no circulation** This effect is related with the concept of added mass. According to [54] Added mass is to be understood as a pressure induced force (and moment), produced by the forced harmonic motion of the body, which is proportional to the acceleration of the body. Another definition is given by [5], where the concept of fluid kinetic energy is used to derive the added mass terms. Tristan and Blanke [5] consider added mass and inertia effects as the reflections of the build up kinetic energy of the fluid as the hull moves through it. The motion of the fluid is associated with the acceleration or deceleration of the hull and it produces the ship to move with an equivalent added mass or inertia, although the fluid does not move with the ship. This concept should not be misunderstood as a finite amount of water connected to the hull, which would be considered as a new body system with additional mass.

- **Motion in an ideal fluid with circulation** Vessel hulls act like airfoils in aerial vehicles, lift and drag forces produced by the hull's displacement in water generate moments and net forces that have to be taken

into account when modelling the craft's behaviour. The hydrodynamic derivatives associated to these forces are proportional to the products uv and ur.

- **Motion in a viscous fluid** This contribution is identified as hydrodynamic damping. The analysis of the motion of vessels in water bodies reveals the presence of hydrodynamic resistance due to a variety of complex interacting phenomena [5]. In Lewis' *"Principles of Naval Architecture"* [65] this effects are analysed and results concerning the hydrodynamic resistance in calm water bodies presented. According to his definition, for a calm water body, the resistance experienced by the vessel movement can be assumed to comprised of these three components:

  - Frictional resistance, due to the motion of the ship's hull in viscous fluid. Two components can be identified that add to this contribution. A linear low-frequency contribution based in the laminar boundary layer theory, and a high frequency one due to turbulent boundary layer theory which is often referred as quadratic or nonlinear skin friction [54].
  - Wave making resistance, created by the energy loss due to the generated waves produced by the hull's motion in water. It is usually referred to as residual resistance, as the main component of viscous motion resistance is made up of frictional resistance and eddy resistance incurs only a small portion of the total. A more in depth analysis of this effect is given in [79].
  - Eddy resistance, produced by the energy carried away by eddies generated from the hull. This effect is caused by Vortex Shedding produced at the end of the hull or the vessel appendages. It is dependent on the Reynolds number and the geometry of the hull. Vortex shedding is the smallest contributing factor to viscous motion resistance.

- **Restoring forces and moments** Buoyancy and gravitational forces are considered restoring forces. In surface vessels the effects of the restoring forces are highly dependent on the locations of the centre of gravity of the vessel and centre of buoyancy and the metacentric height. Restoring forces make important contributions to the stability (roll, pitch and heave) of the vessel.

- **Environmental disturbances** Environmental disturbances cause forces and moments on the hulls. Generally they are divided into wave , wind and ocean current disturbances. Wave disturbances have great impact on motion modelling since in non linear models, hydrodynamic derivatives are dependent on the wave induced oscillation frequency of the body. Wind and ocean current sourced forces and moments are usually linearised and can be modelled with different degrees of complexity.

B.3 Hydrodynamic derivatives

Hydrodynamic derivatives can be treated as an expression of the force produced in a certain direction due to the effects of the linear and angular velocity or speed. One of the most common ways of expressing the hydrodynamic

derivatives is as an expansion of the hull forces as a Taylor series. Taylor series expansions uses partial derivatives to describe the value of a function of multiple variables. For example, a generic function $F(x, y, z)$ with 3 variables would have the following Taylor expansion [80]:

$$\begin{aligned} Y(x,y,z) = & \\ & F(x_0,y_0,z_0) + \frac{\partial F}{\partial x}(x-x_0) + \frac{\partial F}{\partial y}(y-y_0) + \frac{\partial F}{\partial z}(z-z_0) \\ & + \frac{1}{2}\frac{\partial^2 F}{\partial^2 x}(x-x_0)^2 + \frac{1}{2}\frac{\partial^2 F}{\partial^2 y}(y-y_0)^2 + \frac{1}{2}\frac{\partial^2 F}{\partial^2 z}(z-z_0)^2 \\ & + \frac{1}{2}\frac{\partial^2 F}{\partial x \partial y}(x-x_0)(y-y_0) + \frac{1}{2}\frac{\partial^2 F}{\partial x \partial z}(x-x_0)(z-z_0) \\ & + \frac{1}{2}\frac{\partial^2 F}{\partial y \partial z}(y-y_0)(z-z_0) + \dots \end{aligned} \tag{92}$$

The hydrodynamic coefficients can be identified as the partial fractions:

$$F_x = \frac{\partial F}{\partial x} \qquad F_{xx} = \frac{\partial^2 F}{\partial^2 x} \qquad F_{xy} = \frac{\partial^2 F}{\partial x \partial y} \qquad \cdots \tag{93}$$

Linear approaches to hydrodynamic forces modelling through Taylor series expansion, similar to Davidson and Schiff's manoeuvring model [52], only take the linear terms of the series expansion. The expressions are usually simplified thanks to the assumptions made during the analysis process. In Davidson and Schiff's [52] linear model, seen in equation 94), it can be seen that the hydrodynamic derivatives related to surge motion are not present in neither of the expressions of the transversal force Y and yaw moment N due to surge decoupling.

$$Y = Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} + Y_v v + Y_r r + Y_\delta \delta_R \tag{94}$$

$$N = N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} + N_v v + N_r r + N_\delta \delta_R \tag{95}$$

B.3.1 Computation of Hydrodynamic Derivatives.

Hydrodynamic derivative computation presents several methods that can be used to approximate with the higher possible accuracy the values of these coefficients in order to obtain a good model of the vessel. Methods range from fully empirical measurements to theoretical approaches. System identification and recursive parameter estimation along with typical facilities like free oscillators, test tunnels and the Planar Motion Mechanism (PMM) technique are consistently applied successfully to model parametrization. Theoretical methods, in particular potential theory, are commonly used in Naval design software for the calculation of hydrodynamic derivatives of CAD models.

Potential theory stands out as the main theoretical tool for the computation of hydrodynamic derivatives. Potential theory studies fluid motion based on the assumption of ideal fluids (irrotational and incompressible). This analysis neglects the effect of viscous flow along the hull thus dismissing the hydrodynamic effects produced by this phenomenon. However this simplification is usually appropriate for engineering purposes although, in some cases, it is important to

supplement the analysis by adding the effects of viscosity. One example of this is when considering manoeuvring and propeller-rudder-hull interactions [64].

Generally, the potential theory programs that are used to calculate hydrodynamic derivatives provide with results for the added mass and damping matrices using a two dimensional approach to the problem. 2-D potential theory, or strip theory, approximates the motion of the fluid to a two dimensional problem based on the fact that, in slender bodies, cross directional flow variation is greater than its variation in the longitudinal direction of the ship. Strip theory, as its name implies, divides the submerged hull into a finite number of strips and calculates their individual 2-D added mass and damping coefficients. The summation of the contribution of each strip provides a valid 3-D coefficient approximation. Other similar approaches like panel methods, use a direct 3-D approach by integrating the wetted surface of the hull into small discrete elements for coefficient calculations.

Linearised models with estimated hydrodynamic derivatives using the strip theory method suffer from the fact that frictional forces are not computed within this method. However, for most engineering applications this effect can be ignored if precise accuracy is not needed in the model. Besides this, when the angle of attack of the ship's hull with the water is small, viscous separation of the cross flow fluid motion can be assumed to have little significance [10].

Newman's Marine Hydrodynamics [10] provides with strip theory equations for the values of the coefficients along with a comparison analysis between theoretical and experimental values on several real vessels. In the analysis, the source of disparity between the theoretical and empirical values is noted as the viscous friction component not analysed in potential theory. Nevertheless, the approximate values obtained through the theoretical approach give a satisfactory prediction of the hydrodynamic derivatives. Table 12 shows the expressions for the theoretical hydrodynamic derivatives obtained through strip theory. In the table, ρ is the density of the fluid, U is the forward speed of the vessel, T is the draft depth of the hull and L is the hull length. For simplicity draft depth T is assumed to be constant along the length of the hull.

Hydrodynamic derivative	String theory theoretical expression
$Y_{\dot{v}}$	$-\frac{\pi}{2}\rho UT^2$
$Y_{\dot{r}}$	0
$N_{\dot{v}}$	0
$N_{\dot{r}}$	$-\frac{\pi}{24}\rho T^2L^3$
Y_v	$-\frac{\pi}{2}\rho UT^2$
Y_r	$\frac{\pi}{4}\rho UT^2L$
Y_v	$-\frac{\pi}{4}\rho UT^2L$
Y_r	$-\frac{\pi}{8}\rho UT^2L^2$

Table 12: Theoretical expressions for the hydrodynamic derivatives obtained through strip theory. Expressions obtained from [10].

References

[1] A. Farinelli, L. Iocchi, and D. Nardi, "Multirobot systems: a classification focused on coordination," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 5, pp. 2015–2028, 2004.

[2] "Parrot." `https://www.parrot.com`. [accessed 15 February 2017].

[3] D. Borreguero Díaz, "Diseño y desarrollo de una estación base fluctuante para misiones aéreas en entornos fluviales," 2016.

[4] J. Liu and R. Hekkenberg, "Sixty years of research on ship rudders: effects of design choices on rudder performance," *Ships and Offshore Structures*, vol. 0, no. 0, pp. 1–18, 2016.

[5] T. Perez and M. Blanke, *Mathematical ship modelling for control applications*. Ørsted-DTU, Automation, 2002.

[6] B. W. McCormick, A. S. Aljabri, S. J. Jumper, and Z. N. Martinovic, "The analysis of propellers including interaction effects," in *SAE Technical Paper*, SAE International, 02 1979.

[7] O. Osorio, "Drone 2.0 (GrabCad).." Available at: `https://grabcad.com/library/drone-2-0` [accessed 13 September 2016].

[8] T. Bresciani, "Modelling, identification and control of a quadrotor helicopter," *MSc Theses*, 2008.

[9] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, pp. 20–32, Sept 2012.

[10] J. N. Newman, *Marine hydrodynamics*. Cambridge, Mass.: MIT Press, 1977.

[11] R. Wetzel, *Limnology: Lake and River Ecosystems*. Elsevier Science, 2001.

[12] P. A. Soranno, K. S. Cheruvelil, K. E. Webster, M. T. Bremigan, T. Wagner, and C. A. Stow, "Using landscape limnology to classify freshwater ecosystems for multi-ecosystem management and conservation," *BioScience*, vol. 60, no. 6, pp. 440–454, 2010.

[13] M. Saska, J. Mejia, D. Stipanovic, V. Vonasek, K. Schilling, and L. Preucil, "Control and Navigation in Manoeuvres of Formations of Unmanned Mobile Vehicles," *European Journal of Control*, vol. 19, pp. 157–171, March 2013.

[14] J. N. Weaver, D. Z. Frank, E. M. Schwartz, and A. A. Arroyo, "Uav performing autonomous landing on usv utilizing the robot operating system,"

[15] J. Roldán, P. Garcia-Aunon, M. Garzón, J. de León, J. del Cerro, and A. Barrientos, "Heterogeneous multi-robot system for mapping environmental variables of greenhouses," *Sensors*, vol. 16, p. 1018, Jul 2016.

[16] "Mathworks." `https://uk.mathworks.com/`. [accessed 15 February 2017].

[17] M. Ahmadian, Z. Nazari, N. Nakhaee, and Z. Kostic, "Model based design and sdr," in *DSPenabledRadio, 2005. The 2nd IEE/EURASIP Conference on (Ref. No. 2005/11086)*, pp. 8–pp, IET, 2005.

[18] K. Ling, D. Chow, A. Das, and S. L. Waslander, "Autonomous maritime landings for low-cost vtol aerial vehicles," in *2014 Canadian Conference on Computer and Robot Vision*, pp. 32–39, May 2014.

[19] C. Meinig, N. Lawrence-Slavas, R. Jenkins, and H. M. Tabisola, "The use of saildrones to examine spring conditions in the bering sea: Vehicle specification and mission performance," in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–6, Oct 2015.

[20] D. Liu, X. Luan, F. Zhang, J. Jin, J. Guo, and R. Zheng, "An usv-based laser fluorosensor for oil spill detection," in *2016 10th International Conference on Sensing Technology (ICST)*, pp. 1–4, Nov 2016.

[21] C. Flener, M. Vaaja, A. Jaakkola, A. Krooks, H. Kaartinen, A. Kukko, E. Kasvi, H. Hyyppä, J. Hyyppä, and P. Alho, "Seamless mapping of river channels at high resolution using mobile lidar and uav-photography," *Remote Sensing*, vol. 5, p. 6382–6407, Nov 2013.

[22] M. Casado, R. Gonzalez, T. Kriechbaumer, and A. Veal, "Automated identification of river hydromorphological features using uav high resolution aerial imagery," *Sensors*, vol. 15, p. 27969–27989, Nov 2015.

[23] A. S. Woodget, P. E. Carbonneau, F. Visser, and I. P. Maddock, "Quantifying submerged fluvial topography using hyperspatial resolution uas imagery and structure from motion photogrammetry," *Earth Surface Processes and Landforms*, vol. 40, no. 1, pp. 47–64, 2015.

[24] A. Woodget, "Drones for river habitat assessment? a report from the international symposium on ecohydraulics," *BHS Newsletter*, pp. 17–19, Dec 2016.

[25] A. Martins, A. Dias, J. Almeida, H. Ferreira, C. Almeida, G. Amaral, D. Machado, J. Sousa, P. Pereira, A. Matos, V. Lobo, and E. Silva, "Field experiments for marine casualty detection with autonomous surface vehicles," in *2013 OCEANS - San Diego*, pp. 1–5, Sept 2013.

[26] Z. Huang, W. Liu, X. Wang, X. Song, X. Xu, X. Chen, L. Ma, L. Tang, *et al.*, "Design of an intelligent trimaran usv for maritime rescue," in *The 26th International Ocean and Polar Engineering Conference*, International Society of Offshore and Polar Engineers, 2016.

[27] S. Siyang and T. Kerdcharoen, "Development of unmanned surface vehicle for smart water quality inspector," in *2016 13th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1–5, June 2016.

[28] S. Bhattacharya, H. Heidarsson, G. S. Sukhatme, and V. Kumar, "Cooperative control of autonomous surface vehicles for oil skimming and cleanup," in *Robotics and automation (ICRA), 2011 IEEE international conference on*, pp. 2374–2379, IEEE, 2011.

[29] R. R. Murphy, E. Steimle, C. Griffin, C. Cullins, M. Hall, and K. Pratt, "Cooperative use of unmanned sea surface and micro aerial vehicles at hurricane wilma," *Journal of Field Robotics*, vol. 25, no. 3, pp. 164–180, 2008.

[30] A. Matos, A. Martins, A. Dias, B. Ferreira, J. M. Almeida, H. Ferreira, G. Amaral, A. Figueiredo, R. Almeida, and F. Silva, "Multiple robot operations for maritime search and rescue in eurathlon 2015 competition," in *OCEANS 2016 - Shanghai*, pp. 1–7, April 2016.

[31] V. Djapic, C. Prijic, and F. Bogartz, "Autonomous takeoff landing of small uas from the usv," in *OCEANS 2015 - MTS/IEEE Washington*, pp. 1–8, Oct 2015.

[32] A. Mancini, E. Frontoni, P. Zingaretti, and S. Longhi, "High-resolution mapping of river and estuary areas by using unmanned aerial and surface platforms," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 534–542, June 2015.

[33] E. Pinto, F. Marques, R. Mendonça, A. Lourenço, P. Santana, and J. Barata, "An autonomous surface-aerial marsupial robotic team for riverine environmental monitoring: Benefiting from coordinated aerial, underwater, and surface level perception," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pp. 443–450, Dec 2014.

[34] J. Li and Y. Li, "Dynamic analysis and pid control for a quadrotor," in *2011 IEEE International Conference on Mechatronics and Automation*, pp. 573–578, Aug 2011.

[35] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 153–158, Oct 2007.

[36] Y. Sun, "Modeling, identification and control of a quad-rotor drone using low-resolution sensing," 2012.

[37] Q. Li, *Grey-box system identification of a quadrotor unmanned aerial vehicle*. PhD thesis, Citeseer, 2014.

[38] G. Beinset and J. S. Blomhoff, *Controller design for an unmanned surface vessel*. PhD thesis, 2007.

[39] C. R. Sonnenburg and C. A. Woolsey, "Modeling, identification, and control of an unmanned surface vehicle," *journal of Field Robotics*, vol. 30, no. 3, pp. 371–398, 2013.

[40] N. Mijlad, E. Elwarraki, and A. Elbacha, "Simscape electro-thermal modelling of the pin diode for power circuits simulation.," *IET Power Electronics*, vol. 9, no. 7, pp. 1521 – 1526, 2016.

[41] L. Gamage, C. de Silva, and R. Campos, "Design evolution of mechatronic systems through modeling, on-line monitoring, and evolutionary optimization," *Mechatronics*, vol. 22, no. 1, pp. 83 – 94, 2012.

[42] N. A. Dung and A. Shimada, "Humanoid climbing robot modeling in matlab-simechanics," in *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, pp. 4162–4167, Nov 2013.

[43] J. F. de Canete, V. Muñoz-Martinez, J. Luque, J. Barbancho, and J. Rozan, "Modeling and simulation of the short-term arterial pressure control system using an object oriented approach," in *2015 23rd Mediterranean Conference on Control and Automation (MED)*, pp. 117–123, June 2015.

[44] España, "Real decreto 552/2014, de 27 de junio, por el que se desarrolla el reglamento del aire y disposiciones operativas comunes para los servicios y procedimientos de navegación aérea y se modifica el real decreto 57/2002, de 18 de enero, por el que se aprueba el reglamento de circulación aérea. *Boletín Oficial del Estado*, 1 de julio de 2014, núm. 159, pp. 50141 - 50319.." Available at:`www.boe.es/diario_boe/txt.php?id=BOE-A-2014-6856` [accessed 14 February 2017].

[45] S. Bouabdallah, A. Noth, and R. Siegwart, "Pid vs lq control techniques applied to an indoor micro quadrotor," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, pp. 2451–2456 vol.3, Sept 2004.

[46] P. Castillo, R. Lozano, and A. Dzul, "Stabilization of a mini-rotorcraft having four rotors," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, pp. 2693–2698 vol.3, Sept 2004.

[47] B. Erginer and E. Altug, "Modeling and pd control of a quadrotor vtol vehicle," in *2007 IEEE Intelligent Vehicles Symposium*, pp. 894–899, June 2007.

[48] A. Tayebi and S. McGilvray, "Attitude stabilization of a four-rotor aerial robot," in *2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601)*, vol. 2, pp. 1216–1221 Vol.2, Dec 2004.

[49] P. McKerrow, "Modelling the draganflyer four-rotor helicopter," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 4, pp. 3596–3601 Vol.4, April 2004.

[50] N. H. Norrbin, "Theory and observations on the use of a mathematical model for ship manoeuvring in deep and confined waters," tech. rep., DTIC Document, 1971.

[51] K. Nomoto and K. Taguchi, "On steering qualities of ships," *Journal of Zosen Kiokai*, no. 101, pp. 57–66, 1957.

[52] K. Davidson and L. Schiff, "Turning and course keeping qualities of ships," *Transactions of SNAME*, vol. 4, p. 49, 1946.

[53] J. van Amerongen, *Adaptive steering of ships: a model- reference approach to improved manoeuvring and economical course keeping.* PhD thesis, 1982.

[54] T. I. Fossen, *Guidance and control of ocean vehicles.* Chichester ; New York : Wiley, 1994. Includes bibliographical references (p. [455]-474) and index.

[55] P. Mishra, S. Panigrahi, and L. C. S. Das, "Ships steering autopilot design by nomoto model,"

[56] L. Moreira, T. I. Fossen, and C. G. Soares, "Modeling, guidance and control of "esso osaka" model," *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 85 – 90, 2005. 16th IFAC World Congress.

[57] M. Blanke, *Ship Propulsion Losses Related to Automatic Steering and Prime Mover Control*. Technical University of Denmark, 1981.

[58] A. Christensen and M. Blanke, *A Linearized State-space Model of Steering and Roll of a High-speed Container Ship*. Technical University of Denmark (DTU), 1986. Teaching report.

[59] J. van Amerongen and J. C. L. van Cappelle, "Mathematical modelling for rudder roll stabilization," 6th Ship Control Systems Symposium, Ottawa, Canada, 1981.

[60] K. Son and K. Nomoto, "On the coupled motion of steering and rolling of a high speed container ship," *Journal of the Society of Naval Architects of Japan*, vol. 1981, no. 150, pp. 232–244, 1981.

[61] A. Ross, T. Perez, and T. I. Fossen, "A novel manoeuvering model based on low-aspect-ratio lift theory and lagrangian mechanics," *IFAC Proceedings Volumes*, vol. 40, no. 17, pp. 229 – 234, 2007.

[62] A. Ross, "Nonlinear manoeuvring models for ships: a lagrangian approach," 2008.

[63] I. A. F. Ihle, J. Jouffroy, and T. I. Fossen, "Formation control of marine surface craft using lagrange multipliers," in *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 752–758, Dec 2005.

[64] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.

[65] E. Lewis, "Principles of naval architecture: Volume ii-resistance, propulsion, and vibration, vol. 2, 2 edn, soc," *Nav. Archit. Mar. Eng*, 1988.

[66] W. Van Berlekom, "Effects of propeller loading on rudder efficiency," in *Proceedings of the 4th Ship Control Systems Symposium*, October 1975.

[67] I. Martínez, *Termodinamica Basica y Aplicada*. Dossat, 1992.

[68] P. H. Miller, "Dynamic lift coefficients for spade rudders on yachts," tech. rep., DTIC Document, 2007.

[69] M. Araki, "Pid control," *Control Systems, Robotics and Automation: System Analysis and Control: Classical Approaches II*, pp. 58–79, 2009.

[70] W. K. Ho, O. P. Gan, E. B. Tay, and E. L. Ang, "Performance and gain and phase margins of well-known pid tuning formulas," *IEEE Transactions on Control Systems Technology*, vol. 4, pp. 473–477, Jul 1996.

[71] W. Tan, J. Liu, T. Chen, and H. J. Marquez, "Comparison of some well-known pid tuning formulas," *Computers and Chemical Engineering*, vol. 30, no. 9, pp. 1416 – 1423, 2006.

[72] B. Herissé, T. Hamel, R. Mahony, and F.-X. Russotto, "Landing a vtol unmanned aerial vehicle on a moving platform using optical flow," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 77–89, 2012.

[73] D. Lee, T. Ryan, and H. J. Kim, "Autonomous landing of a vtol uav on a moving platform using image-based visual servoing," in *2012 IEEE International Conference on Robotics and Automation*, pp. 971–976, May 2012.

[74] N. Jeurgens, "Implementing a simulink controller in an ar. drone 2.0," 2016.

[75] F. A. Administration, *Helicopter Flying Handbook*. Skyhorse Publishing Company, Incorporated, 2013.

[76] Z. S. Spakovszky, *Thermodynamics and Propulsion*. Massachusetts Institute of Technology: MIT, 2006.

[77] A. Hajivand and S. Mousavizadegan, "The effect of memory in passive nonlinear observer design for a dp system," in *Proceedings of the Dynamic Positioning Conference*, 2010.

[78] O. Faltinsen, *Sea loads on ships and offshore structures*, vol. 1. Cambridge university press, 1993.

[79] J. V. Wehausen, "The wave resistance of ships," vol. 13 of *Advances in Applied Mechanics*, pp. 93 – 245, Elsevier, 1973.

[80] A. B. of Shipping, *Guide for Vessel Maneuverability*. American Bureau of Shipping, 2006.